



**ROBERT GORDON
UNIVERSITY•ABERDEEN**

OpenAIR@RGU

The Open Access Institutional Repository at Robert Gordon University

<http://openair.rgu.ac.uk>

Citation Details

Citation for the version of the work held in 'OpenAIR@RGU':

| |
|---|
| <p>XIANG, G., 2012. Automatic 3D facial modelling with deformable models. Available from <i>OpenAIR@RGU</i>. [online]. Available from: http://openair.rgu.ac.uk</p> |
|---|

Copyright

Items in 'OpenAIR@RGU', Robert Gordon University Open Access Institutional Repository, are protected by copyright and intellectual property law. If you believe that any material held in 'OpenAIR@RGU' infringes copyright, please contact openair-help@rgu.ac.uk with details. The item will be removed from the repository while the claim is investigated.

AUTOMATIC 3D FACIAL MODELLING
WITH DEFORMABLE MODELS

GUOFU XIANG

PhD

2012

This page is intentionally left blank.

AUTOMATIC 3D FACIAL MODELLING WITH DEFORMABLE MODELS

GUOFU XIANG

A thesis submitted in partial fulfilment of the
requirements of the
Robert Gordon University
for the degree of Doctor of Philosophy

School of Computing Science and Digital Media
IDEAS Research Institute
Robert Gordon University

December 2012

Automatic 3D Facial Modelling with Deformable Models

by Guofu Xiang

Copyright © 2012 Guofu Xiang. All rights reserved.

Supervisors:

Prof. Patrik O'B. Holt

Dr. Xiangyang Ju

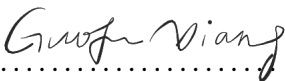
Location:

Aberdeen, UK

Declaration

I declare that this thesis was composed entirely by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

Aberdeen, UK, December 2012

Signed: 

GUOFU XIANG

Date: 12/12/2012

This page is intentionally left blank.

Abstract

Facial modelling and animation has been an active research subject in computer graphics since the 1970s. Due to extremely complex biomechanical structures of human faces and people's visual familiarity with human faces, modelling and animating realistic human faces is still one of greatest challenges in computer graphics.

Since we are so familiar with human faces and very sensitive to unnatural subtle changes in human faces, it usually requires a tremendous amount of artistry and manual work to create a convincing facial model and animation. There is a clear need of developing automatic techniques for facial modelling in order to reduce manual labouring. In order to obtain a realistic facial model of an individual, it is now common to make use of 3D scanners to capture range scans from the individual and then fit a template to the range scans. However, most existing template-fitting methods require manually selected landmarks to warp the template to the range scans. It would be tedious to select landmarks by hand over a large set of range scans. Another way to reduce repeated work is synthesis by reusing existing data. One example is expression cloning, which copies facial expression from one face to another instead of creating them from scratch.

This aim of this study is to develop a fully automatic framework for template-based facial modelling, facial expression transferring and facial expression tracking from range scans. In this thesis, the author developed an extension of the iterative closest points (ICP) algorithm, which is able to match a template with range scans in different scales, and a deformable model, which can be used to recover the shapes of range scans and to establish correspondences between facial models. With the registration method and the deformable model, the author proposed a fully automatic approach to reconstructing facial models and textures from range scans without re-

quiring any manual interventions. In order to reuse existing data for facial modelling, the author formulated and solved the problem of facial expression transferring in the framework of discrete differential geometry. The author also applied his methods to face tracking for 4D range scans. The results demonstrated the robustness of the registration method and the capabilities of the deformable model. A number of possible directions for future work were pointed out.

Keywords: registration, deformable model, template fitting, facial expression transferring, surface reconstruction, texture reconstruction, facial expression tracking, range scans

Dedicated to my family and friends.

This page is intentionally left blank.

Acknowledgements

I would like to express my deep sense of gratitude to my supervisors, Prof. Patrik O'B. Holt and Dr. Xiangyang Ju, for their invaluable guidance, support and encouragement throughout my study. Special thanks go to Dr. Eyad Elyan for his patient guidance on refinements to the thesis and the results. I have learned a great deal about becoming a researcher.

I would like to thank Prof. Ian Allison, Prof. Susan Craw, Prof. Dawei Song, Prof. John McCall, Dr. Ines Arana, Dr. Andrei Petrovski, Mr. Martin Simpson, and Dr. Virginia Dawod for their great support for my research. I am thankful to the school admin and IT team for their kind support.

I would like to thank all the people in CTC, to name a few, Stella Asimwe, Rahman Mukras, Sandy Brownlee, Amandine Orecchioni, Thierry Mamer, Ulises Cervino Beresi, Ibrahim Adeyanju, Olivier Regnier-Coudert, Ben Horsburgh, Nuka Nwiabu, Richard Akinlofa, Leszek Kaliciak, Peter Milne, and Ibrahim Alwawi. It was an enjoyable experience working with them. Special thanks go to David Lee, Jean-Claude Golovine, and Malcolm Clark, who started the unforgettable journey along with me.

I would like to thank Peng Zhang, Yanghui Wu, Jun Wang, Lei Wang, and Miki Sun for having many wonderful leisure activities together, such as cooking, travelling, and sporting, which make my leisure life better.

I am indebted to Prof. Qingchuan Zhang, Prof. Hong Miao, and Prof. Xiaoping Wu in USTC for their immense support for my research. I would like to thank all the people in the PhotoMech Lab., Pengtao Cao, Zhengyu Miao, Zheyang Guo, Shengcheng Jin, and others, for the delightful experience of working in the lab.

I am grateful for the help from my classmates and friends: Xiaoyan Chen, Di Kong, Wei Li, Peng Gao, Jian Zhang, Shizhong Bao, Xiaoli Wu, Yali Hu, Qingxiang Tan, Qiong Wu, and others.

Lastly, but most importantly, I would like to thank my family for supporting me throughout this study. My father Changfa Xiang, my mother Yuedi Wang and my elder brothers Guoping Xiang and Guoxiang Xiang provide a foundation in my life on which I know I can always depend.

Guofu Xiang
Aberdeen, UK
December 2012

Contents

| | |
|---|-----------|
| Contents | xi |
| List of Figures | xv |
| 1 Introduction | 1 |
| 1.1 Aims and Objectives | 3 |
| 1.2 Thesis Overview | 4 |
| 2 Literature Review | 5 |
| 2.1 Historical Overview | 5 |
| 2.2 Facial Modelling and Animation Techniques | 9 |
| 2.2.1 Parameterised Facial Models | 9 |
| 2.2.2 Physically-Based Facial Models | 10 |
| 2.2.3 Performance-driven Facial Models | 12 |
| 2.2.4 Example-based Facial Models | 14 |
| 2.2.5 Summary | 17 |
| 2.3 Registration Techniques | 18 |
| 2.3.1 Rigid Registration | 19 |
| 2.3.2 Non-rigid Registration | 22 |
| 2.3.3 Summary | 23 |
| 2.4 Deformation Techniques | 24 |
| 2.4.1 Deformable Models | 24 |
| 2.4.2 Surface-Based Deformation | 25 |
| 2.4.3 Deformation Transfer | 25 |
| 2.4.4 Summary | 26 |
| 2.5 Face Tracking Techniques | 27 |
| 3 Facial Model Reconstruction | 29 |
| | xi |

| | | |
|----------|---|-----------|
| 3.1 | Overview | 29 |
| 3.2 | Dataset | 31 |
| 3.3 | Data Preprocessing | 32 |
| 3.4 | Registration | 33 |
| 3.5 | Deformable Model | 35 |
| 3.6 | Surface Reconstruction | 40 |
| 3.7 | Texture Reconstruction | 41 |
| 3.8 | Facial Expression Tracking | 41 |
| 3.9 | Results and Discussion | 43 |
| 3.10 | Chapter Summary | 50 |
| 4 | Facial Expression Transferring | 51 |
| 4.1 | Overview | 52 |
| 4.2 | Facial Expression Representation | 54 |
| 4.3 | Displacement Deformation Gradients | 55 |
| 4.4 | Deformation Reconstruction | 58 |
| 4.5 | Guidance Deformation Gradients | 60 |
| 4.5.1 | Correspondences | 60 |
| 4.5.2 | Local Transformation | 62 |
| 4.6 | Results and Discussion | 64 |
| 4.7 | Chapter Summary | 67 |
| 5 | Evaluation | 69 |
| 5.1 | Registration | 69 |
| 5.1.1 | FRGC Dataset | 70 |
| 5.1.2 | Comparison | 72 |
| 5.2 | Deformable Model | 72 |
| 5.2.1 | The Curve of the RMS Error | 72 |
| 5.2.2 | Influence of the Control Parameters k_s and k_b | 75 |
| 5.2.3 | Deformation | 75 |
| 5.3 | Facial Expression Transferring | 76 |
| 5.4 | Chapter Summary | 78 |
| 6 | Conclusions and Future Work | 79 |
| 6.1 | Achievements and Contributions | 79 |
| 6.2 | Future Work | 80 |

| | |
|--|------------|
| A Triangle Meshes | 83 |
| A.1 Simplicial Complexes | 83 |
| A.2 Topological and Geometrical Realisations | 84 |
| A.3 Piecewise Linear Basis Functions | 85 |
| A.4 Two-manifold and Neighbourhoods | 86 |
| B Discrete Differential Operators | 87 |
| B.1 Discrete Potential Fields | 87 |
| B.2 Discrete Gradient Fields | 88 |
| B.3 Discrete Gradient Operator | 88 |
| B.4 Discrete Divergence Operator | 90 |
| B.5 Discrete Laplace-Beltrami Operator | 93 |
| B.6 Laplacian Smoothing | 95 |
| C FRGC Test Results | 97 |
| D Qualitative Evaluation Questionnaire | 109 |
| Bibliography | 127 |

This page is intentionally left blank.

List of Figures

| | | |
|------|--|----|
| 2.1 | Evolution of facial modelling and animation | 7 |
| 2.2 | Parke’s parameterised facial model | 10 |
| 2.3 | Physically-based facial model | 11 |
| 2.4 | Anatomically accurate head model | 12 |
| 2.5 | Performance-driven facial model | 12 |
| 2.6 | Live facial puppetry | 13 |
| 2.7 | High-quality passive facial performance capture | 14 |
| 2.8 | Morphable model | 15 |
| 2.9 | Example-based facial rigging | 16 |
| 2.10 | An illustration of the ICP algorithm | 19 |
| 2.11 | The Digital Michelangelo project | 20 |
| 2.12 | Robust rigid registration | 21 |
| 2.13 | Human body shape reconstruction | 22 |
| 2.14 | Deformation transfer | 26 |
| 2.15 | Markerless 3D face tracking | 28 |
| 3.1 | Overview of facial model reconstruction | 30 |
| 3.2 | A sample range scan of the FRGC v2.0 dataset | 31 |
| 3.3 | Data preprocessing of raw range scans | 32 |
| 3.4 | An illustration of surface deformation | 36 |
| 3.5 | A visual plot of the sparse matrix of the deformable model | 39 |
| 3.6 | FED 4D range scans | 42 |
| 3.7 | The RMS error curve of a registration | 43 |
| 3.8 | The influence of initial scales | 44 |
| 3.9 | The influence of initial relative rotations | 46 |
| 3.10 | A close-up of the deformation in the nose region | 47 |
| 3.11 | The deformation process of the template | 47 |

| | | |
|------|--|----|
| 3.12 | The result of the texture reconstruction | 48 |
| 3.13 | The result of facial model reconstruction | 49 |
| 3.14 | Results of facial expression tracking (female) | 50 |
| 3.15 | Results of facial expression tracking (male) | 50 |
| 4.1 | The problem of facial expression transferring | 53 |
| 4.2 | Illustration of the displacement representation | 54 |
| 4.3 | Examples of the displacement representation | 55 |
| 4.4 | Issues with directly transferring expression displacements | 56 |
| 4.5 | The linear system of discrete displacement deformation gradients | 57 |
| 4.7 | Illustration of correspondences | 62 |
| 4.8 | Illustration of local transformation | 63 |
| 4.9 | Results of facial expression transferring (head models) | 65 |
| 4.10 | Results of facial expression transferring (reconstructed models) | 66 |
| 5.1 | Examples of the FRGC range scans containing outliers | 70 |
| 5.2 | A successful registration for the FRGC dataset | 71 |
| 5.3 | A failed registration for the FRGC dataset | 71 |
| 5.4 | Registration of the synthetic data of a foot | 73 |
| 5.5 | Registration of two range scans of an owl | 73 |
| 5.6 | Registration of two range scans of a frog | 73 |
| 5.7 | The curve of the RMS error for a deformation process | 74 |
| 5.8 | The influence of the control parameter k_s | 75 |
| 5.9 | The influence of the control parameter k_b | 76 |
| 5.10 | An example deformation process | 76 |
| 5.11 | Overall rating for facial expression transferring | 77 |
| 5.12 | Rating for each facial expressions | 78 |
| A.1 | Mesh representation | 84 |
| A.2 | Piecewise linear basis function | 85 |
| B.1 | Linear basis functions | 89 |
| B.2 | Discrete gradient operator | 90 |
| B.3 | Discrete divergence operator | 92 |



Introduction

The human face plays an important role in human life, owing to the fact that it can not only be used as a form of biometric information to identify an individual, but also be capable of producing a large variety of facial expressions that supply important visual information for communication. In computer graphics, it has been a long-standing interest to mimic human faces and expressions in computers using a variety of facial models, such as parameterised facial models (Parke, 1972, 1974, 1982; Parke and Waters, 2008), physically-based facial models (Platt and Badler, 1981; Waters, 1987; Terzopoulos and Waters, 1990; Lee et al., 1993, 1995; Sifakis et al., 2005, 2006), performance-driven facial models (Williams, 1990; Pighin et al., 1998; Weise et al., 2009, 2011), and example-based facial models (Blanz and Vetter, 1999; Blanz et al., 2003; Pyun et al., 2003; Li et al., 2010; Seol et al., 2011). However, due to the diversity of human faces and the complexity of the underlying mechanisms of facial motions, creating realistic facial models and animations turns out to be a very difficult task. Although immense efforts have been devoted to this area, there is still no computational system available that approximates the performance of real human faces (Griesser et al., 2007; Orvalho et al., 2012).

Modelling and animating realistic face models is a substantial challenge in computer graphics, especially for facial expressions, because humans are capable of identifying unnatural behaviour, due to their everyday familiarity and sensitivity to facial appearance. For this reason, it usually requires a tremendous amount of artistry and manual work to creating a convincing

facial model and animation. For example, the *Digital Emily* project (Alexander et al., 2009) took 3 artists, 2 animators and 3 technicians almost 7 months to create a realistic digital actor. There is a clear need for more automatic techniques to reduce the painstaking work.

With the advances in 3D scanning techniques (Curless, 2000; Bernardini and Rushmeier, 2002; Lanman and Taubin, 2009; Coudrin et al., 2011), it has become popular to create realistic facial models by using 3D scanners to capture highly detailed facial shapes from real faces (Zhang et al., 2004; Weise et al., 2009, 2011; Beeler et al., 2011). A common approach for facial model reconstruction from range scans is to fit a template to the range scans (Li et al., 2009). Generally, there are two problems involved in the template-fitting process: one is how to globally align a template with a range scan and another is how to deform the aligned template onto the range scan. Most existing template-fitting methods make use of manually selected landmarks to warp the template to the range scan and control the deformation of the template by a weighted cost function consisting of the terms of different constraints (Allen et al., 2003; Sumner and Popović, 2004; Weise et al., 2009, 2011). However, it would be tedious to select corresponding landmarks by hand on a large set of range scans. In addition, it would be more plausible to deform the template based on a physically based deformable model. This thesis will present a novel fully automatic template-fitting approach to reconstructing both the shapes and the textures of range scans using a deformable model without requiring any manual interventions.

On the other hand, synthesis by reusing existing data is a desirable feature to reduce repeated work in computer graphics (Noh and Neumann, 2001; Pyun et al., 2003; Sumner and Popović, 2004; Botsch and Sorkine, 2008; Ben-Chen et al., 2009; Seol et al., 2012). Noh and Neumann (2001) first proposed an approach for expression cloning, which copies facial expressions from one face to another instead of creating them from scratch. In this approach, each expression is encoded as the vertex displacements between the neutral and expression faces, and the displacements are locally adjusted on a new face with the aid of heuristic search. Sumner and Popović (2004) generalised this approach for transferring the arbitrary deformation between two triangle meshes. They represented the deformation as the affine transformations of corresponding triangles rather than the vertex displacements. In contrast, this thesis will formulate and solve the problem of facial expres-

sion transferring in the framework of discrete differential geometry, where facial expressions are encoded as deformation gradients.

Recently, 3D face tracking for spacetime range scans has drawn a considerable amount of attention in computer graphics and vision Zhang et al. (2004); Yin et al. (2008); Weise et al. (2009, 2011). In contrast to face tracking in 2D images, 3D face tracking generally employs both geometry and texture information of a sequence of faces, which could improve the tracking accuracy. For 3D face tracking, a template is often used to track facial motions in the frame-to-frame manner, and optical flow is used to enhance template tracking by establishing inter-frame correspondences from images DeCarlo and Metaxas (2000); Zhang et al. (2004); Weise et al. (2009, 2011). This thesis will present the work on automatic facial expression tracking for 4D range scans using the proposed registration method and deformable model.

1.1 Aims and Objectives

This work aims to develop an automatic framework for template-based facial modelling, facial expression transferring and facial expression tracking from range scans. The following are the major objectives of this research.

- Firstly, a new registration algorithm needs to be developed in order to align a template with range scans in different sizes.
- Secondly, this research needs to develop a novel deformable model based on thin-shells. This deformable model should be able to control surface deformation through adjusting a small number of parameters.
- Thirdly, this work needs to propose an automatic surface reconstruction algorithm for recovering the shapes and texture of range scans, based on the developed registration method and deformable model.
- Next, this research will study a new approach for transferring facial expressions for one facial model to another.
- Finally, a new spacetime tracking method will be proposed to track facial the expressions exhibited in a sequence of range scans.

1.2 Thesis Overview

The rest of this thesis is organised as follows. Firstly, Chapter 2 gives a literature review of related research work including face modelling, registration, deformable modelling, and face tracking. Then, the work of this thesis is presented in Chapters 3 and 4. In Chapter 3, the work of facial model and texture reconstruction from range scans is presented, including pre-processing of range scans, registration of a template onto range scans in different scales, and deformation of the aligned template toward target range scans. Chapter 4 shows the work of facial expression transfer between facial models, where facial expressions are encoded in a scale-invariant representation and can be faithfully reconstructed from such a representation. Next, the quantitative and qualitative evaluations of the proposed methods are discussed in Chapter 5. Finally, the achievements and contributions of this research are listed in Chapter 6. Some future research directions are pointed out.

Literature Review

This chapter presents a literature review of related work to this research, mainly focusing on facial modelling and animation, registration, surface deformation and face tracking. It is organised as follows. Section 2.1 first describes a brief historical overview of this area. Then, the following sections review various related techniques. Section 2.2 presents a review of various facial modelling and animation techniques proposed in the last 40 years. rigid and non-rigid registration techniques for aligning 3D shapes with data are discussed in Section 2.3. Next, Section 2.4 reviews various deformation techniques, including deformable models, surface-based deformation and deformation transfer. Finally, Section 2.5 presents a review of face tracking techniques.

2.1 Historical Overview

Facial expression analysis has been the subject of scientific investigation for over one hundred years. The study of facial movements and expressions started from a biological point of view. It was first suggested by Darwin (1872) that facial expressions are innate and consist of habitual movements that depend on emotions and the state of the mind. In the past, facial expression analysis was primarily a research subject for psychologists. The goal was to find the relationship between emotions and facial expressions. Ekman and Friesen (1971) demonstrated six basic emotions, comprising happiness, sadness, fear, disgust, surprise and anger, that each possesses a dis-

tinctive content together with a unique facial expression. They seem to be universal across human ethnicities and cultures. Through the years, a number of systems have been developed to describe movements underlying facial expressions. One of the most well-known representational systems is the Facial Action Coding System (FACS) by Ekman and Friesen (1978). It decomposes facial expressions into dozens of small facial movement units, called action units (AUs), which correspond to single or multiple facial muscle activations. The combination of action units produces expressions. With later advances made in related areas, such as face detection (Hjelmås and Low, 2001; Zhang and Zhang, 2010), face tracking (Yilmaz et al., 2006; Chellappa et al., 2011) and face recognition (Zhao et al., 2003; Wright et al., 2009), the research subject has become automatic facial expression analysis, which involves face acquisition, facial feature extraction, and facial expression classification. Interested readers should refer to the recent comprehensive surveys (Fasel and Luetttin, 2003; Fang et al., 2012).

The earliest work on facial modelling and animation in computer graphics started in the 1970s. Parke developed the first parameterised 3D facial model in 1974 (Parke, 1974; Parke and Waters, 2008). Using photogrammetric techniques, he collected 3D data of facial expressions from real faces and created facial animations by interpolating between the facial expressions. From then on, computer facial modelling and animation has become an active research area in computer graphics. New facial modelling approaches and advanced animation techniques have been developed. Figure 2.1 shows an overview of the evolution of facial animation since the 1970s until today.

With the motivation of creating realistic facial expressions, several muscle based facial models were proposed in the 1980s. Platt and Badler (1981) extended Parke's work by utilising simulation of muscles rather than interpolation to animate a face. Waters (1987) developed a muscle-based facial model, which allows to control the simulation of skin deformation by a limited number of parameters. Magnenat-Thalmann et al. (1988) proposed the concept of abstract muscle action procedure for controlling human face animation and synchronising speech.

In the 1990s, Terzopoulos and Lee constructed an anatomically motivated facial model based on scanned data, and endowed it with a layered mass spring system driven by muscle contractions (Terzopoulos and Waters, 1990; Lee et al., 1993, 1995). It was later extended by Kähler et al. (2001).

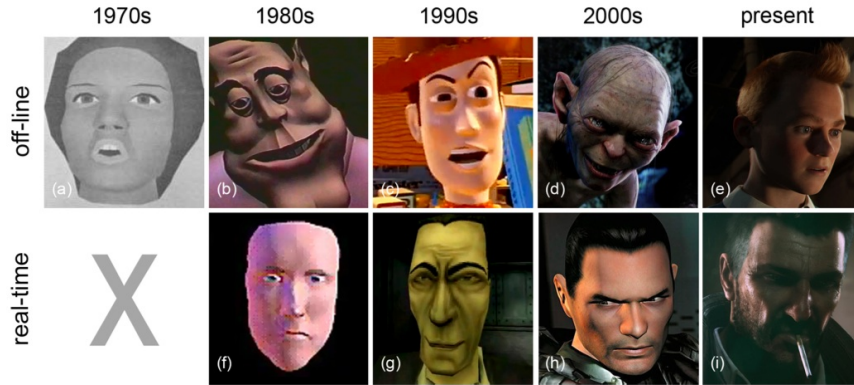


Figure 2.1: Evolution of facial modelling and animation: (a) Parke’s parametric facial model (1972); (b) Tony de Peltrie (1985), the first short animated film to use a parameterised facial model; (c) Toy Story (1997), the first CGI film; (d) Gollum (2004), realistic CGI character in a leading role in a live action film; (e) The Adventures of Tintin (2011), the current state of the art with performance capture in an animation film; (f) Mike and Talking Head (1988), first real-time virtual puppetry; (g) Half-Life (1998), early facial animation on 3D video-games; (h) Doom 3 (2004), bone-based facial rig for video games; (i) The Samaritan Demo (2011), the current state of the art in facial animation in real-time. by courtesy of (Orvalho et al., 2012).

Meanwhile, performance-driven facial animation was first developed by Williams (1990), and was later extended by Guenter et al. (1998); Pighin et al. (1998). Performance capture techniques use the recorded motion of an actor to drive a performance of a virtual character, most often from a set of tracking motion capture markers attached to the actor’s face. In the late 1990s, DeCarlo et al. (1998) used variational modelling and facial anthropometry techniques to construct smooth facial models. At the same time, tools supporting 3D facial animation decoders with different degrees were standardized in MPEG-4 (Ostermann, 1998; Abrantes and Pereira, 1999; Lavagetto and Pockaj, 1999).

Since Blanz and Vetter (1999) first introduced a morphable facial model, example-based facial models have become popular in the 2000s. The morphable facial model allows to create new characters based on a 3D face database and to modify facial attributes by varying the model coefficients. Later, Blanz et al. (2003) extended the morphable facial model for face recognition across variation in pose and a wide range of illuminations. Pyun et al.

(2003) proposed a example-based approach for facial expression cloning (Noh and Neumann, 2001) from a source model to a target model while preserving the characteristic features of the target model. Li et al. (2010) introduced a method for generating facial blendshape rigs from a set of example poses of a CG character. Many techniques for speech synchronization and non-verbal speech-related facial expressions were developed (Kalberer and Gool, 2001; Albrecht et al., 2002; Deng and Neumann, 2006; Sifakis et al., 2006). The statistical analysis of facial expressions has been accomplished by Abboud et al. (2003); Terzopoulos et al. (2004).

Recently, with the rapid advance of hardware, realtime high-quality performance driven facial animation has played a more significant role in computer graphics. Wand et al. (2009) described a system for reconstructing a single shape and its deformation from a temporal sequence of point clouds from real-time 3D scanner data. Weise et al. (2009, 2011) developed a markerless live puppetry system using a real-time scanner. They demonstrated that high-quality real-time facial expression capture and transfer is possible without costly studio infrastructure, face markers, or extensive user assistance. Beeler et al. (2011) proposed an approach of high-quality passive and markerless facial performance capture based on anchor frames. This approach uses stereo reconstruction to establish a high resolution triangle mesh that is propagated through the entire performance. (Seol et al., 2012) formulated the retargeting problem as a Poisson equation and developed a spacetime facial animation retargeting method for blendshape face models.

However, despite the recent technical advance within this area, the currently available techniques are still not able to meet the requirement from many envisaged applications (Ersotelos and Dong, 2008). To compromise with limited computation resources, considerable amount of simplification has to be made in anatomy-based facial models. Also, due to the limitation from the current image and video analysis, the performance driven approach has to involve a large number of equipments and the accuracy is still subjective to further enhancement. Therefore, facial modelling and animation is still an on-going research issue, and there is a long way before satisfactory completion of the technology.

2.2 Facial Modelling and Animation Techniques

Modelling and animating realistic facial has challenged researchers in computer graphics since its beginning. Since Parke's pioneering work (Parke, 1972, 1974), there have been extensive efforts on the development of facial modelling and animation with a variety of facial models proposed. Those facial models generally fall into the following four categories: parameterised facial models (Parke, 1972, 1974, 1982), physics-based facial models (Platt and Badler, 1981; Waters, 1987; Terzopoulos and Waters, 1990; Lee et al., 1993, 1995; Sifakis et al., 2005, 2006), performance-driven facial models (Williams, 1990; Pighin et al., 1998; Weise et al., 2009, 2011), and example-based facial models (Blanz and Vetter, 1999; Blanz et al., 2003; Pyun et al., 2003; Li et al., 2010; Seol et al., 2011). A detailed overview of facial modelling and animation can be found in (Parke and Waters, 2008; Ersotelos and Dong, 2008).

2.2.1 Parameterised Facial Models

The first attempt at facial modelling and animation involved key frame animation (Parke, 1972). This basic idea of this approach is that a sequence of 3D facial shapes (key frames) is completely specified and the facial shapes in between these key frames are computed by interpolation. However, at that time, the immense shape space of human faces made this approach impractical, because it was difficult to build a large number of key frames of 3D facial shapes (Parke and Waters, 2008).

This motivated Parke and others to develop parameterised facial models, which generated facial shapes through an appropriate set of parameters (Parke, 1974, 1982; Platt and Badler, 1981). Figure 2.2 shows some examples of Parke's facial model. Using these model, animators can create facial expressions by interpolating the parameter values rather than facial shapes. The parameters control various facial features, such as lip opening height, width, and protrusion. Combinations of these parameters provide a large range of facial expressions with relatively low computational costs.

However, there are certain limitations of parameterised facial models. One is that there is no systematic way to arbitrate between two conflicting parameters to blend expressions that affect the same vertices. Unless animators specify the shape parameters with care, these models will pro-

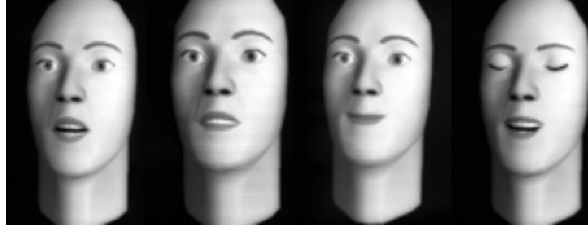


Figure 2.2: Parke's parameterised facial model (Parke, 1982).

duce incorrect shapes, unrealistic motions, and other spurious effects. For this reason, facial parameterisations are designed to only affect specific facial regions (Parke, 1982), however this often introduces noticeable motion boundaries. Another limitation of parameterised facial models is that control parameters are often coupled with the shape of a specific model. It is difficult to reuse these parameters for other facial models without tedious manual tuning.

Nowadays, advances in 3D scanners have made the high-quality shape acquisition of human faces increasingly more feasible (Lanman and Taubin, 2009; Coudrin et al., 2011; Beeler et al., 2011). Thus, data-driven key frame animation has once again become an option for facial animation (Zhang et al., 2004; Weise et al., 2009, 2011), and has been widely used in film projects to animate virtual characters, such as *The Lord of the Rings*, *King Kong* and *Avatar* (Lewis and Anjyo, 2010).

2.2.2 Physically-Based Facial Models

The motivation of physically-based facial models is to create synthetic facial expressions by simulating facial muscle actions on a facial model. Usually, physically-based facial models represent human faces with hierarchical structures and incorporate physically-based approximations to anatomical structures such as facial tissues and muscles.

Platt and Badler (1981) first used a mass-spring system to simulate the skin and muscles. The mass-spring system propagates muscle forces in an elastic spring mesh that models skin deformation. The goal was to drive the face at a higher level of control. Extensions and improvements to this technique have been reported in (Waters, 1987; Terzopoulos and Waters, 1990; Lee et al., 1993, 1995).

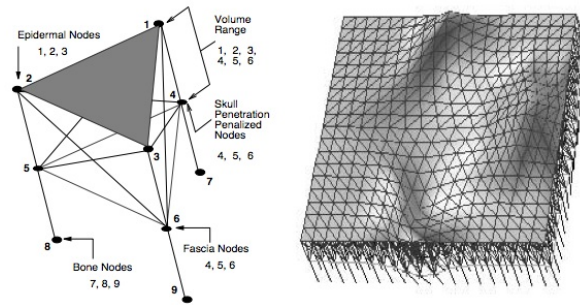


Figure 2.3: Three connected mesh layers of the physically-based facial model (Lee et al., 1995).

Further, Waters (1987) represented the action of muscles using primary motivators on a non-specific deformable topology of the face. The muscle actions themselves are tested against FACS (Facial Action Coding System) (Ekman and Friesen, 1978) which employs action units directly to one muscle or a small group of muscles. This muscle-based method deforms a facial mesh using motion fields in delineated regions of influence.

Magenat-Thalmann et al. (1988) defined a model where the action of a muscle is simulated by a procedure, called an Abstract Muscle Action procedure (AMA), which acts on the vertices composing the human face figure. It is possible to animate a human face by manipulating the facial parameters using AMA procedures. By combining the facial parameters obtained by the AMA procedures in different ways, we can construct more complex entities corresponding to the well-known concept of facial expression.

Because the human face consists of a biological tissue layer with nonlinear deformation properties, a muscle layer knit together under the skin, and an impenetrable skull structure beneath the muscle layer. Taking all of these structures into account, anatomically-based methods either extend mass-spring structures into three connected mesh layers (Figure 2.3) to model anatomical facial behaviour (Terzopoulos and Waters, 1990; Lee et al., 1993, 1995), or build a high resolution, anatomically and bio-mechanically accurate flesh and muscle model (Figure 2.4) of a subject's face (Sifakis et al., 2005, 2006). This physical simulation propagates the muscle forces through the physics-based synthetic skin thereby deforms the skin to produce facial expressions. It significantly improves the realism of synthetic facial expressions compared to the earlier techniques Parke (1982); Platt and Badler

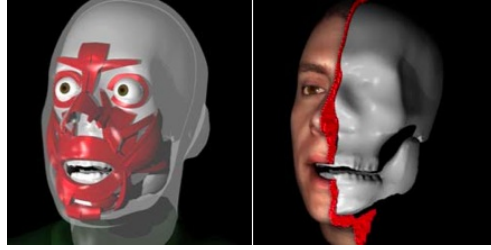


Figure 2.4: Anatomically accurate head model (Sifakis et al., 2005).

(1981); Waters (1987); Magnenat-Thalmann et al. (1988). This is, of course, subject to an increase in computational expense.

2.2.3 Performance-driven Facial Models

From a practical viewpoint, an ideal input device for driving facial animation is probably the real human face that mimics desired expressions. This motivates the development of performance-driven facial animation. Performance-driven approaches usually use a motion capture system to capture the facial motions of live actors' performances, and then reconstruct photorealistic 3D animation of the captured expressions.

There are many efforts that have been done in this area. Williams first introduced the term—performance driven facial animation—to the computer graphics community (Williams, 1990). He tracked the 2D face motions of a performer from a single video stream. Later, Guenter et al. (1998) extended this approach to 3D recovery from multiple video streams (Figure 2.5). Pighin et al. (1998) developed a technique for creating photorealistic textured 3D facial models from photographs of a human subject, and

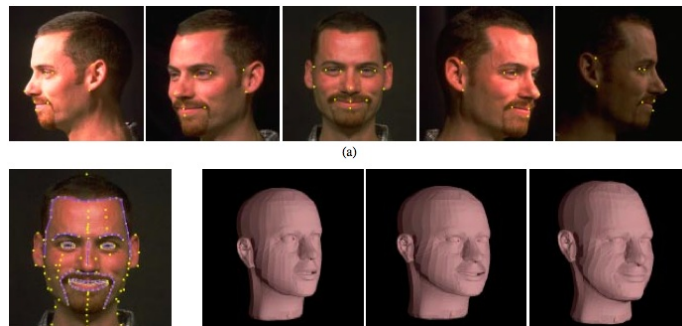


Figure 2.5: Performance-driven facial model (Guenter et al., 1998).



Figure 2.6: Live facial puppetry (Weise et al., 2009).

for creating smoothing transitions between different facial expressions by morphing between these different models. Zhang et al. (2006) developed a geometry-driven facial expression synthesis system for creating synthesized expression images with photorealistic and natural looking details from a given set of feature points. Weise et al. (2009, 2011) proposed a real-time facial tracking system with transfer to another person's face (Figure 2.6). They built personalised templates with 24 manually selected landmarks and then used these templates to track facial expressions captured at 15 fps subject to various constraints, such as optical flow constraints (DeCarlo and Metaxas, 2000).

Recently, Beeler et al. (2011) presented a performance-driven markerless facial capture system that uses high resolution per-frame geometry acquisition using stereo geometry reconstruction (Figure 2.7). Also in 2011 Huang et al. (2011) introduced a new approach that combines motion capture data with a minimal set of face scans in a blend shape interpolation framework, which allows the reconstruction of high-fidelity 3D facial performances. In a more practical way, Weise et al. (2011) described a real time method for animating a face using tracking algorithm that combines 3D geometry and 2D texture registration to achieve motion to a rig based blend shapes.

With the captured motion data from performers' faces, the resulting animated faces are far more natural and lifelike. This results in the popularity of performance-driven facial models in many entertainment areas, such as game and film industries (Pighin, 2006). However, when there are con-



Figure 2.7: High-quality passive facial performance capture (Beeler et al., 2011).

siderable differences in the facial shapes between a live performer and an animated character, expression mapping between them becomes a difficult problem. Moreover, in order to faithfully reproduce the facial expressions in performers' faces, a large number of feature markers have to be attached onto the faces (Williams, 1990; Guenter et al., 1998). Manual placement of markers can be tedious and invasive. It has still been a challenge to accurately track these markers (Walder et al., 2009). Furthermore, the markers must be digitally removed from the videos if face colour or texture is to be acquired. Also, the marker resolution is naturally limited, and detailed pore-scale performance capture has not been demonstrated with this approach. In contrast, markerless performance driven facial animation (Zhang et al., 2004; Weise et al., 2009, 2011; Beeler et al., 2011) requires less manual setup.

2.2.4 Example-based Facial Models

Example-based facial modelling approaches concern creating facial model through the combination of existing models. Such methods require support from a collection of facial models, which usually are captured from a variety of real faces using 3D scanners (Curless, 2000; Bernardini and Rushmeier, 2002; Lanman and Taubin, 2009; Coudrin et al., 2011). Given desired facial features, optimisation method is used to find the right combination coefficients. The linear combination of the collected facial models with these optimised coefficients provides a close match between the synthetic model and the desired facial features. The prominent work of these methods was published in the literature (Blanz and Vetter, 1999; Blanz et al., 2003; Pyun et al., 2003; Allen et al., 2003; Vlasic et al., 2005; Li et al., 2010).



Figure 2.8: Morphable model (Blanz and Vetter, 1999).

Blanz and Vetter (1999) presented a facial modelling technique named as morphable modelling (Figure 2.8). A distinct strength of this modelling approach is that it allows generate a wide variety of facial modelling with minimal user input. More specifically, a facial model can be created from a single photograph supplied by users and the created model matches the facial features portrayed by the photograph. This method requires an example set of 3D facial models. Morphable facial modelling is based on transforming the shapes and textures of these example facial models into a vector space representation. The shape and texture of a new facial model is represented by a linear combination of these transformed vectors. However, the implementation of the morphable facial modelling is not straightforward, since it requires a large collection of 3D facial models within which the dense point to point correspondence between the models have to be established.

Pyun et al. (2003) developed example-based approach for cloning facial expressions of a source model to a target model while reflecting the characteristic features of the target model in the resulting animation. In this approach, a set of key-models for source and target models are first constructed as a preprocess; then the target key-models are parameterised using the source key-models and the weight functions for the parameterised target key-models are predefined based on radial basis functions; finally, for a given input model with some facial expressions, the output model is obtained by blending the target key-models with weight values evaluated for the target key-models. However, if an expression falls too far outside from

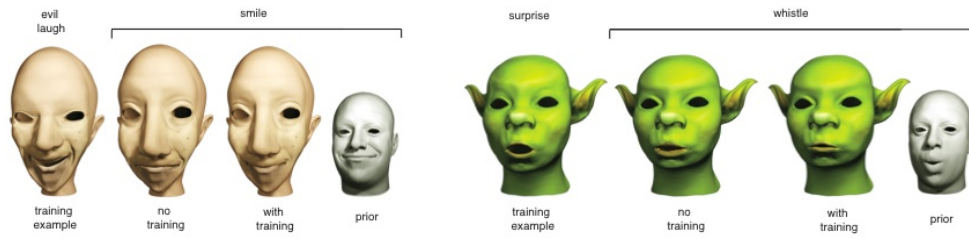


Figure 2.9: Example-based facial rigging (Li et al., 2010).

the basic of the constructed source key-models, this method cannot correctly copy that expression.

While there are several flavours of bilinear and multi-linear (also called tensor) models, they can generally be considered as extensions to principal component analysis (PCA), which was used to analyse facial variations in (Banz and Vetter, 1999; Banz et al., 2003). Whereas PCA captures all the variations in a single set of data, multi-linear models can be used to capture independent high-level meta-dimensions of the data. Vlastic et al. (2005) applied this idea to facial cross-mapping, with one dimension capturing the variation in appearance across individuals (between source and target in particular) and a second dimension covering facial expression variation. Allen et al. (2003) applied a similar technique to range scan data for creating a new pose model.

Recently, Li et al. (2010) introduced a method for generating facial blendshape rigs from a set of example poses of a CG character (Figure 2.9). In this approach, a predefined blendshape rig of a generic face is used as a prior to determine the semantics of each blendshape expression. Their system transfers controller semantics and expression dynamics from the generic face to the target blendshape model by solving an optimal reproduction of the training poses. This approach enables a scalable design process, where the user can iteratively add more training poses to refine the blendshape expression space.

The advantages of example-based facial models contain that, (a) they allow for human control of feature attributes in resulting images and animations; (b) they do not need to identify facial features. However, these models need support of a database of facial models and are generally confined by the shape space of available faces and expressions in the database.

Table 2.1: Comparison between four categories of facial models

| Facial models | Major characteristics |
|----------------------------------|--|
| Parameterised facial models | Ideas: generate facial shapes through a set of parameters that control facial features. Advantages: low computational costs. Disadvantages: issues with conflicting parameters; hard to reuse. |
| Physically-base facial models | Ideas: generate facial shapes through simulating the physical properties of facial skin and muscles. Advantages: realistic facial motions. Disadvantages: high computational cost. |
| Performance-driven facial models | Ideas: animate facial models through the motion data captured from actors' performances. Advantages: natural and lifelike results. Disadvantages: need feature markers; labour-intensive. |
| Example-based facial models | Ideas: create new facial models from range scans or images that are captured from real faces. Advantages: allow for the control of feature attributes; easy to model specific individuals. Disadvantages: be confined by the shape space of range scans; require a database for shape analysis. |

2.2.5 Summary

Comparing above parameterised facial models, physically-based facial models, performance driven facial models and example-based facial models (Table 2.1), we can see a number of strengths and weaknesses of these models:

- Parameterised facial models need a generic model, which can generate facial expressions by adjusting a set of parameters with relatively low computational costs. However, these models have issues with conflicting parameters and it is difficult to reuse those control parameters for a different model.
- Physically-based facial models simulate the action of the underlying facial muscles and tissues and thus can produce realistic facial expressions. However, the physics-based simulation is usually a time-consuming process and requires a lot of computational resources.
- Performance driven facial models can produce natural and lifelike fa-

cial animation and has been widely used in film projects, but marker-based approaches can be tedious and invasive to setup. For markerless performance driven facial animation, it is still a challenge to eliminate “swimming” artifacts (Beeler et al., 2011), which are motions in the tangent plane.

- Example-based facial models allow for human control of facial features in resulting facial models, but these models require a database of facial models for shape analysis and are usually confined by the shape space of that database.

2.3 Registration Techniques

Registration is a fundamental problem in computer vision and computer graphics (van Kaick et al., 2011). Generally, it is a process of overlaying two or more images or range scans of the same scene taken at different times, from different viewpoints, or by different sensors. This section mainly considers registration techniques for shape modelling. For image registration, interested readers should refer to related surveys (Maintz and Viergever, 1998; Zitová and Flusser, 2003; Shams et al., 2010) and the book by Szeliski (2011).

Surface registration is an essential component of the 3D model acquisition pipeline (Bernardini and Rushmeier, 2002; Rusinkiewicz et al., 2002; Botsch et al., 2007). Due to limitations of 3D scanning technology, typically multiple range scans must be acquired to cover the whole object’s surface from different coordinate system. To allow them to be recombined to reconstruct the surfaces that represent the original objects, the individual range images must be aligned, or registered, into a common coordinate system. Furthermore, registering templates to a set of deforming surfaces provides cross-parameterisation (Kraevoy and Sheffer, 2004), and facilitates deformation transfer (Sumner and Popović, 2004), shape interpolation (Alexa et al., 2000), and statistical shape analysis (Loncaric, 1998).

Surface registration may consider rigid or non-rigid shapes. The former assumes that two (or more) surfaces are related by a rigid transformation. The latter allows deformation between them.

2.3.1 Rigid Registration

Rigid registration is a challenging problem (Tam et al., 2012). Firstly, the data itself poses many difficulties, which may include noise, outliers, and limited amounts of overlap. Noise may take the form of perturbations of points, or unwanted points close to a 3D surface. Outliers are unwanted points far from the surface, which can seriously affect results if not discarded. Limited overlap arises due to different parts of the object being in view in each scan; typically the number of scans is kept low for efficiency, with few points in common between successive scans. Further problems may arise due to self-occlusion when the object is scanned from certain viewing angles. While such problems can be mitigated by carefully scanning, they are hard to avoid completely. Secondly, variations in initial positions and orientations, as well as resolutions of data, can also affect algorithm performance.

The iterative closest point (ICP) algorithm (Figure 2.10), first introduced by Besl and McKay (1992) and Zhang (1992), is the most popular method for rigid registration due to its simplicity and low computational complexity. It has become the dominant method for aligning 3D models based purely on the geometry of the shapes. The algorithm is widely used for registering the outputs of 3D scanners. ICP starts with two shapes and an initial estimate for their relative rigid-body transformation. It then iteratively refines the transformation by alternating the steps of selecting corresponding points on the shapes, and finding the best rotation and translation that minimises an error metric based on the distance between the corresponding points.

Many variants of ICP have been proposed that affect all phases of the

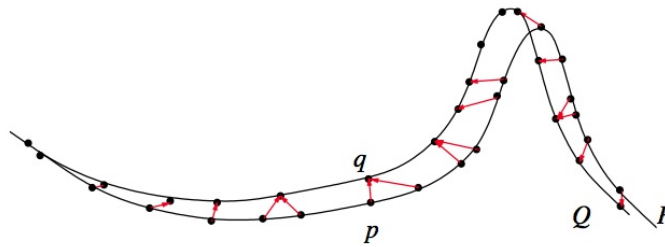


Figure 2.10: An illustration of the ICP algorithm, by courtesy of (Bernardini and Rushmeier, 2002). Point matches are defined based on shortest Euclidean distance. Scan P is then transformed to minimise the length of the displacement vectors, in the least-squares sense.

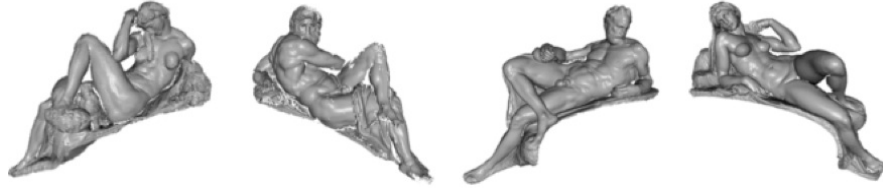


Figure 2.11: The Digital Michelangelo project (Levoy et al., 2000).

algorithm from the selection and matching of points to the minimisation strategy (Rusinkiewicz and Levoy, 2001). When pairwise registration is used sequentially to align multiple views errors accumulate, and the global registration is far from optimal. Turk and Levoy (1994) used a cylindrical scan that covers most of the surface of the object, and then incrementally register other scans to it. Pulli (1999) described another incremental multiview registration method, which first aligns scans pairwise with each other and uses the pairwise alignments as constraints that the multiview step enforces while evenly diffusing the pairwise registration errors. Pulli's method is particularly suitable for registering large datasets and it was used in the Digital Michelangelo Project (Levoy et al., 2000) to digitise large statues (Figure 2.11). Gelfand et al. (2003) improved the alignment performance and quality of the ICP algorithm by selecting feature points that would constrain all degrees of freedom in the rigid-body transformation.

The initial positions of the point sets tremendously influence the final results of the ICP algorithm, because the first correspondence is derived from this initial configuration. Thus, a crucial step in ICP-based methods is to estimate the initial alignment of the shapes. Generating the initial alignment may be done by a variety of methods, such as tracking scanner position, spin-image surface signatures (Johnson and Hebert, 1999), computing principal axes of scans (Dorai et al., 1997), exhaustive search for corresponding points (Chen et al., 1998), or user input. Gelfand et al. (2005) presented a global registration algorithm that aligns two shapes without any assumption about their initial positions (Figure 2.12). This algorithm uses a volume descriptor to locate the feature points and the potential corresponding points, which bring the two shapes into a coarse alignment, and then a conventional ICP algorithm can be used for fine registration. oshua Podolak et al. (2006) suggested pre-alignment based on the reflectional symmetry axes of the shapes is another as another effective solution.



Figure 2.12: Robust rigid registration (Gelfand et al., 2005).

To overcome the ICP limitations, many probabilistic methods were developed Rangarajan et al. (1997); Gold et al. (1998); Luo and Hancock (2001); Chui and Rangarajan (2003); Myronenko and Song (2010). These methods consider the alignment of two point sets as a probability density estimation problem, where one point set represents the Gaussian Mixture Model (GMM) centroids and the other one represents the data points, and use soft assignment of correspondences that establishes correspondences between all combinations of points according to some probability which generalises the binary assignment of correspondences in ICP. It was shown in (Chui and Rangarajan, 2003) that alternating soft assignment of correspondences and transformation is equivalent to the Expectation Maximisation (EM) algorithm (Moon, 1996) for GMM with E-step to compute the probabilities of correspondences and M-step to update the transformation. These probabilistic methods perform better than conventional ICP, especially in the presence of noise and outliers (Myronenko and Song, 2010).

Another class of rigid registration methods is the spectral methods. Scott and Longuet-Higgins (1991) introduced a noniterative algorithm to associate points of two arbitrary patterns, exploiting some properties of Gaussian proximity matrix of point sets. The algorithm works well with translations, shearing, and scaling deformations, but performs poorly with non-rigid transformation. Ho et al. (2007) proposed a noniterative algorithm for 2D affine registration by searching for the roots of the associated polynomials. Unfortunately, this method does not generalise to higher dimensions. Belongie et al. (2002) introduced the “shape context” of the the point set and thus helps to recover the correspondence between the point sets.

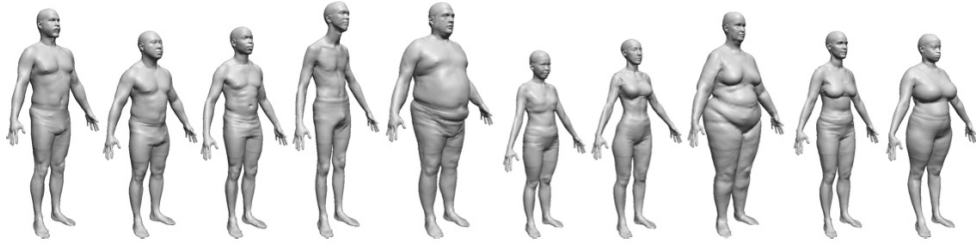


Figure 2.13: Human body shape reconstruction from range scans (Allen et al., 2003).

2.3.2 Non-rigid Registration

Non-rigid registration is even more difficult than rigid registration, as it not only faces the challenges of rigid registration but also needs to account for deformation, so solution space is much larger. Unlike the rigid case, where a few correspondences are sufficient to define one candidate rigid transformation for hypothesis testing, both deformation and alignment in the non-rigid case, without strong prior assumptions, often require a lot more reliable correspondences to define. Establishing meaningful and natural correspondences, however, is a challenging problem in its own right. Choice of appropriate representation for the deformation, and suitable tools for evaluation of non-rigid registration methods are two difficult problems (Tam et al., 2012).

The ICP method can also be used for non-rigid alignment by modifying some of its components. One set of methods computes a weighted correspondence where each assignment has an associated confidence value. These values are crucial for robust outlier detection. After rigidly aligning the shapes with the weighted ICP, the shapes are non-rigidly deformed into each other by computing a warp function such as thin-plate splines (Bookstein, 1989). Hähnel et al. (2003) extended ICP to allow deformable objects to be aligned by computing rigid-body warps on different parts of a joint skeleton. Allen et al. (2003) used an affine transformation at each vertex of the source mesh to allow non-rigid registrations of full-body scans to a high-resolution template (Figure 2.13). However, this approach introduces too many unknown variables (12 unknown variables per each vertex), which make the convergence speed slow. Brown and Rusinkiewicz (2007) extended the use of the thin-plate spline to global non-rigid registration of 3D range

scans. They first recovered sparse feature correspondences between views using a locally weighted, stable ICP computation, then obtained a consistent set of range scans using thin-plate spline warps. In addition to thin-plate splines, radial basis functions (RBFs) (Carr et al., 2001) are also known to be well suited for scattered data interpolation problem and has been used to conform generic models to 3D scanned data with several landmarks (Ju and Siebert, 2001).

Recently, variants of ICP have also been utilised in the context of time-varying surface reconstruction to align the geometry of adjacent time frames (Wand et al., 2007, 2009; Li et al., 2009, 2012; Pekelny and Gotsman, 2008). If a sufficient number of scans is acquired per unit of time, it can be assumed that only small changes take place in the spatial configuration of the shapes (i.e. rigid-body components can be consistently tracked), and so the initial alignment of each frame is not a strong issue in the registration.

Another popular non-rigid registration method is by Chui and Rangarajan (2003). This approach uses Thin Plate Spline (TPS) (Bookstein, 1989) parameterisation of the transformation, following Robust Point Matching (RPM) (Gold et al., 1998), which results in the TPS-RPM method. Similarly to the rigid case, this method uses deterministic annealing and alternate update for soft assignment and TPS parameters estimation. Rather than assigning features bases closest points as in ICP, each pair of points is assigned a probability of corresponding based on a Gaussian function of their distance from each other (softassign).

2.3.3 Summary

The techniques of rigid registration are used to find the optimal rigid-body alignment of two 3D shapes with their initial alignment roughly known or estimated by other approaches. The two 3D shapes are often similar or have overlapping parts. The methods of non-rigid registration generally combine rigid registration with a warping function that can locally fit one shape to another. The quality of the results often relies on the robustness of detecting and matching feature points on both shapes. In this thesis, the work of reconstructing the shapes of facial scans using a template can be also viewed as the non-rigid registrations of the template to the facial scans. However, there are some differences in shapes and scales between the template and the facial scans. This causes difficulties in applying those above techniques to fit

the template to the scans. Instead, the template is first globally aligned with the scans by an extension of ICP with a consideration of the scale problem, and then is gradually deformed onto the scans by a deformable model.

2.4 Deformation Techniques

This section presents the related deformation techniques for manipulating the shapes of objects with emphases on the surface-based deformation and deformation transfer.

2.4.1 Deformable Models

Deformable models primarily represent the use of elasticity theory at the physical level for simulating elastic bodies that respond naturally to applied forces and constraints. Typically, a deformable model involves deformation energy functions and external potential energy functions. The deformation energy functions are usually defined in terms of the geometric degrees of freedom and often include terms that constrain the smoothness of the model. These functions increase monotonically as the model deforms away from a specified natural or rest shape. In the Lagrangian settings the deformation energies give rise to elastic forces internal to the model. The external potential energy functions are often defined in terms of the data of interest to which the model is to be fitted. These potential energies give rise to external forces, which deform the model such that it fits the data.

Since the seminal work by Terzopoulos et al. (1987) on elastically deformable models, deformable models have seen various applications in the fields of medical image analysis (McInerney and Terzopoulos, 1996) and computer graphics (Gibson and Mirtich, 1997; Nealen et al., 2006). Kass et al. (1988) proposed active contour models, called snakes, for interactively matching planar contours to the locations and shapes of object boundaries in images by means of energy minimisation. Terzopoulos and Fleischer (1988) used physically based deformable models to create realistic animations involving the interaction of deformable models with various applied forces, ambient media, and impenetrable obstacles in a simulated physical world. Terzopoulos et al. (1988) made use of deformable models to recover 3D shapes and non-rigid motions. Deformable models were also used in

many other key areas of computer graphics, such as cloth animation and fluid simulation (Nealen et al., 2006; Yu et al., 2012).

2.4.2 Surface-Based Deformation

Surface-based deformation has been a rather new active research area in recent years (Botsch and Pauly, 2006; Botsch et al., 2007; Botsch and Sorkine, 2008). The goal of this research area is to develop fast, robust, and intuitive deformation techniques that provide physically plausible and aesthetically pleasing surface deformations while preserving geometric details.

Generally, surface-based deformation methods are classified into two categories: minimisation of deformation energies and manipulation of differential properties (Botsch and Kobbelt, 2004; Botsch et al., 2006; Botsch and Pauly, 2006; Botsch et al., 2007; Botsch and Sorkine, 2008; Fröhlich and Botsch, 2011). The former is based on the minimisation of the elastic thin-shell energy that measures stretching and bending deformations (Terzopoulos et al., 1987; Botsch and Kobbelt, 2004; Botsch et al., 2006). The latter edits differential surface properties instead of spatial coordinates, such as gradients and Laplacians. For example, the gradient-based editing methods (Sumner and Popović, 2004; Sumner, 2005; Yu et al., 2004; Zayer et al., 2005) deform a surface by prescribing a target gradient field and finding a new surface that matches this gradient field in the least squares sense. Similarly, the Laplacian surface editing methods (Lipman et al., 2004; Sorkine et al., 2004) manipulate the Laplacians of a surface instead of its gradient field.

All those approaches above are usually implemented by simplifying or linearising the problem into a linear system. Linear methods are fast and robust, because the associate linear systems are usually sparse and can be solved efficiently. However, due to the inherent non-linearity of the surface deformation problem, linear methods may yield artefacts especially when the surface deformation contains large rotations (Botsch and Sorkine, 2008).

2.4.3 Deformation Transfer

Deformation transfer refers to the deformation techniques that reproduce the deformation exhibited by one shape on another different shape. The concept of deformation transfer comes with recent increasing demands for reusing existing data and reducing repeated work (Sumner, 2005).

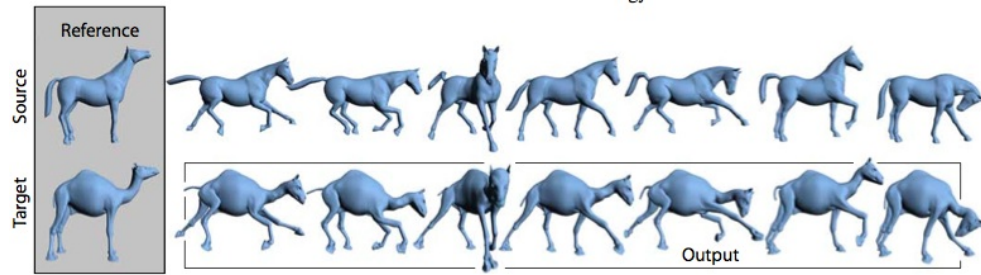


Figure 2.14: Deformation transfer (Sumner and Popović, 2004)

Many approaches have been proposed to transfer the deformation from one mesh to another. Noh and Neumann (2001) first proposed the concept of expression cloning with the idea of taking advantage of existing animation data to retarget them to new models instead of creating new facial animations from scratch. This approach encodes each expression in form of vertex motion vectors and then adapts the directions and the scales of the motion vectors on a new model with aid of a heuristic correspondence search. Sumner and Popović (2004) generalised the idea of expression cloning to a framework for transferring deformations from one triangle mesh to another, and extended it for their mesh-based inverse kinematics technique (Sumner et al., 2005). In this framework (Figure 2.14), the source and target deformations are represented as affine transformations. The known source deformation are mapped to the target via the correspondence map and the target deformation is obtained by solving a constrained optimisation that matches the source transformation as closely as possible while maintaining consistency constraints. Botsch et al. (2006) combined the advantages of deformation transfer and multi-resolution decomposition to achieve detail-preserving surface editing.

2.4.4 Summary

Deformable models has been proved to be useful tools, not only for simulating elastic objects involving natural interactions with the environment, but also for reconstructing shapes and motions of objects in a scene (Zhang et al., 2003, 2004). The formulation of deformable models is generally based on the dynamics of deformable models under the influence of applied forces, which involves time-dependent simulations. However, in a modelling application,

one is typically not interested in a dynamic time-dependent simulation but, instead, directly solves for the equilibrium state of the deformation process. Surface-based deformation techniques described above fall into this kind. The deformable model developed in Chapter 3 is also based on this idea.

The work of facial expression transfer, presented in Chapter 4, is similar to those methods of expression cloning and deformation transfer. In contrast to those methods, this thesis formulates the problem of facial expression transfer within the framework of differential geometry, where a facial expression is represented as the gradient field of the expression face with respect to the neutral face, rather than the motion vertices or the affine transformations. This formulation provides more insights into the problem.

2.5 Face Tracking Techniques

There is a vast body of work on tracking the human face with applications ranging from motion capture to facial expression analysis (Williams, 1990; Basile and Blake, 1998; Yilmaz et al., 2006; Fasel and Luetten, 2003; Weise et al., 2009, 2011). Generally, face tracking algorithms can be classified into two categories: model-free and model-based tracking algorithms.

The model-free tracking algorithms (Williams, 1990; Shi and Tomasi, 1994; Bourel et al., 2000; DeCarlo and Metaxas, 2000) are general purpose point trackers without knowledge of the object. Each facial feature point is usually tracked by performing a local search for the best matching position, or by computing the motion field between two adjacent frames using optical flow techniques (DeCarlo and Metaxas, 2000). However, the model-free methods are susceptible to the inevitable tracking errors due to noise and occlusion.

The model-based methods, on the other hand, focus on explicit modelling the shape of the objects. Extensive work has been focused on the shape representation of deformable objects, such as active contour models (Snakes) (Kass et al., 1988), deformable template methods (Zhang et al., 2004; Wang et al., 2008; Walder et al., 2009), and active appearance models (Cootes et al., 2001; Cristinacce and Cootes, 2006). Although the model-based methods utilize much knowledge about human faces to realise an effective tracking, these models are limited to some common assumptions, *e.g.*, a nearly frontal view face and moderate facial expression changes, and

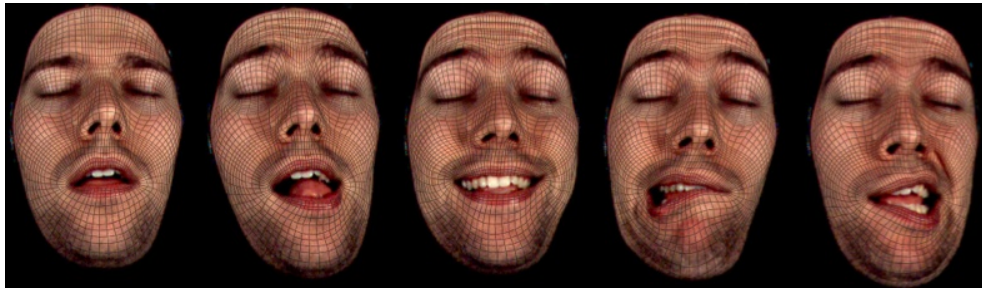


Figure 2.15: Markerless 3D face tracking (Walder et al., 2009)

tend to fail under large pose variations or facial deformations in real world applications.

Recently, advances in real-time 3D capture have enabled the acquisition of dynamically deforming shapes at sustained “video” rates (Yin et al., 2008; Weise et al., 2011). This is attracting increasing attention to realtime tracking 3D shapes acquired over time while moving and deforming. Pekelnny and Gotsman (2008) presented an algorithm for time-varying 3D surface reconstruction. This method segments the input set of point clouds to rigid components, detects rigid motions between the frames, and tracks the skeleton of the subject. Weise et al. (2009, 2011) developed a system for real-time performance-based facial animation, in which realtime tracking was achieved by combining geometry and texture registration with pre-recorded animation priors.

Facial Model Reconstruction

This chapter presents the work of facial model reconstruction from range scans, partial results of which have been published in the paper (Xiang et al., 2010). Section 3.1 presents an overview of facial model reconstruction. Sections 3.2 and 3.3 show the dataset of range scans to be reconstructed and the data pre-processing steps applied on the range scans. Section 3.4 presents an improved ICP algorithm for aligning a template to a range scans using a similarity transformation instead of rigid transformation. Section 3.5 presents a deformable model based on the elastic theory of thin-shells for smoothly deforming a source surface towards a target surface. Sections 3.6 and 3.7 show the details of the improved registration method and the deformable model to reconstruct the shape and the texture of range scans. Section 3.9 presents the results obtained by the proposed methods. Finally, Section 3.10 summarises the facial model reconstruction approach.

3.1 Overview

This chapter presents an automatic approach to creating a realistic facial model from a range scan using a facial template. Figure 3.1 illustrated the automated facial model reconstruction approach. First, the facial template is globally aligned with the range scan such that the scale, rotation, and translation are adjusted to fit the facial template to the range scan. Then the template is smoothly deformed into the range scan in order to capture the

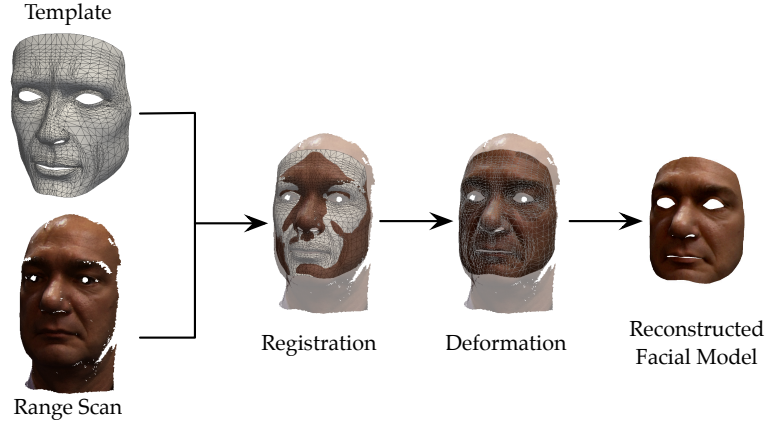


Figure 3.1: Overview of facial model reconstruction: A template is first globally matched with a range scan by using a registration method, and then is smoothly deformed toward the range scan by using a deformable model. The reconstructed facial model has the shape of the range scan and the topological structures of the template.

local geometric details of the range scan. Finally, the reconstructed facial model adapted the shape of the range scan and the topological structures of the template.

The alignment between the facial template and the range scan is a registration problem, which is a fundamental problem in shape acquisition and shape modelling (Gelfand et al., 2005). As reviewed in Section 2.3, there are many approaches to aligning two 3D shapes, such as the ICP variants (Besl and McKay, 1992; Zhang, 1992; Rusinkiewicz and Levoy, 2001), geometric hashing (Wolfson and Rogoutsos, 1997; Lamdan and Wolfson, 1988), and generalised Hough transform (Ballard, 1981). However, classical rigid registration methods usually only compute the rigid transformation (rotation and translation) between two similar shapes with overlaps. In addition to that, we need also to compute the scale of transformation here, because the scale of the range scan is usually different from the facial template. Thus, in order to solve the registration problem here, we need to compute the similarity transformation (scale, rotation and translation) between the template and the range scan. In Section 3.4, an improved version of the ICP algorithm is proposed to solve this scale problem.

In order to smoothly deform the registered template into the range scan, a deformable model is developed in Section 3.5, based on the elastic theory

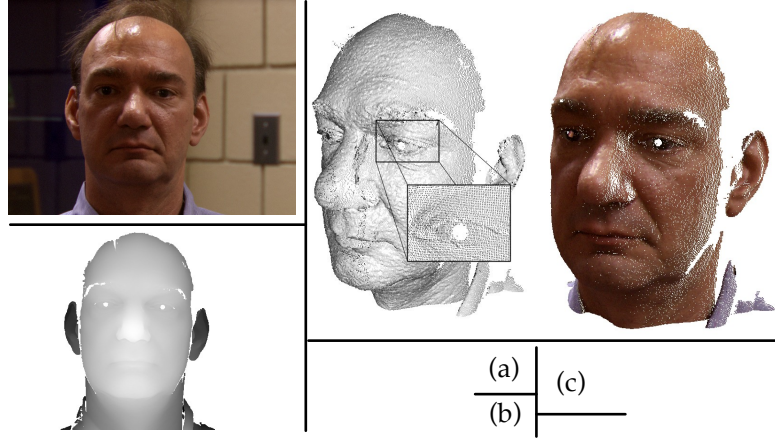


Figure 3.2: A sample range scan of the FRGC v2.0 dataset: (a) 2D texture image; (b) depth image; (c) point cloud in 3D space.

of thin-shells. The deformation of the template is driven by the attractive forces yielded from the corresponding points of the range scans, but is constrained by the internal forces to the stretching and bending deformation energies of the template. The smooth deformation is achieved through iteratively refining the corresponding points and gradually decreasing the resistance of the internal forces.

3.2 Dataset

In this thesis, the range scans to be reconstructed are from the Face Recognition Grand Challenge (FRGC) v2.0 dataset (Phillips et al., 2005). The FRGC v2.0 dataset provides a large number of highly detailed 3D facial scans with texture images. A 3D facial scan in the FRGC v2.0 dataset is actually a 2.5D scan. Figure 3.2 shows a sample scan. For each scan, it has a 2D texture image and a depth image (See Figures 3.2 (a) and (b)). The depth image, also called a point cloud, contains the distance information from the camera sensor to the face. Together the depth and texture images provide a highly detailed 3D face image of the subject (Figure 3.2 (c)). The average number of the points of a scan is about 110K. As shown in Figure 3.1, the template facial model is a triangle mesh of a generic face that has 1880 points and 3580 triangles.

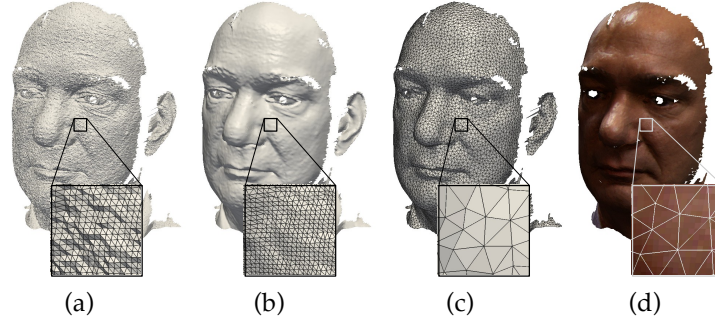


Figure 3.3: Data preprocessing of raw range scans: (a) triangulation; (b) Laplacian smoothing; (c) decimation; (d) cleanup.

3.3 Data Preprocessing

As shown in Figure 3.2, the raw scan is a noisy point cloud with a large number of points, which is inefficient for computation. Thus, the following data preprocessing steps are applied on raw range scans.

Triangulation The first preprocessing step is to reconstruct a triangular surface of a given point cloud. There are many techniques available for converting a point cloud to a triangle mesh (Lorensen and Cline, 1987; Hoppe et al., 1992). However, since the point clouds in the FRGC v2.0 dataset are actually depth images, which have implicit topological structure information about neighbouring points, we simply make use of this information to create triangle meshes by connecting neighbouring points with triangular facets. Figure 3.3 (a) shows the generated triangle mesh after triangulation of the point cloud as shown in Figure 3.2.

Smoothing A triangle mesh generated by the above approach typically shows high frequency noise, *i.e.*, small perturbations in the vertex positions, which do not correspond to shape features. In order to obtain a high-quality surface and to ensure numerical robustness, it is crucial to smooth out these artefacts intrinsic to range scans in such a way that the global shape, or low frequency components, is preserved. In this thesis, the Laplacian smoothing algorithm (Botsch and Pauly, 2006) is used for mesh smoothing (for more details about Laplacian smoothing, please refer to Section B.6). Figure 3.3

(b) shows the result triangle mesh after applying 10 iterations of Laplacian smoothing with the diffusion constant μ set to 0.25.

Decimation A triangle mesh generated from a range scan often contains a large number of points and tiny useless triangles. To improve computation efficiency, it is necessary to reducing the number of triangles while minimally perturbing the shape. In this thesis, an implementation of the edge collapse algorithm (CGAL; Garland and Heckbert, 1997; Lindstrom and Turk, 1998) is used for the purpose of decimation. This process reduces the number of edges to 5% of edges. Figure 3.3 (c) shows a decimated triangle mesh, which still preserves salient facial features while the number of points and triangles is only a small fraction of that of the original mesh.

Cleanup Those unconnected parts are removed. Figure 3.3 (d) shows the result after cleaning up.

3.4 Registration

Given a template and a range scan, let us refer the template as the source triangle mesh \mathcal{M}_s and the range scan the target triangle mesh \mathcal{M}_t (See Appendix A for the formal description of triangle meshes). The goal of the registration here is to find the optimal similarity transformation including the scale $s \in \mathbb{R}^+$, the rotation $\mathbf{R} \in \text{SO}(3)$, and the translation $\mathbf{t} \in \mathbb{R}^3$, such that the target mesh and the transformed source mesh by the optimal transformations are fitted as closely as possible. To solve this optimisation problem, the general framework of ICP variants (Rusinkiewicz and Levoy, 2001) is used. In the following text, an iteration is denoted by a superscript in parentheses.

Initially, the initial similarity transformation $(s^{(0)}, \mathbf{R}^{(0)}, \mathbf{t}^{(0)})$ is automatically estimated from the source mesh \mathcal{M}_s and the target mesh \mathcal{M}_t (See below). The initial point set $\mathcal{P}^{(0)}$ with n points is sampled from the source mesh \mathcal{M}_s , *i.e.*,

$$\mathcal{P}^{(0)} = \{\mathbf{p}_i^{(0)} \in \mathcal{M}_s, i = 1, \dots, n\}. \quad (3.1)$$

Then, the following steps are looped in order to obtain the next estimation of the similarity transformation $(s^{(j+1)}, \mathbf{R}^{(j+1)}, \mathbf{t}^{(j+1)})$ from the current

estimation $(s^{(j)}, \mathbf{R}^{(j)}, \mathbf{t}^{(j)})$ (All the transformations are with respect to the initial point set $\mathcal{P}^{(0)}$):

1. Apply the current estimation of the transformation $(s^{(j)}, \mathbf{R}^{(j)}, \mathbf{t}^{(j)})$ to the initial point set $\mathcal{P}^{(0)}$ and obtain the updated point set $\mathcal{P}^{(j+1)}$:

$$\mathcal{P}^{(j+1)} = \{\mathbf{p}_i^{(j+1)} = s^{(j)} \mathbf{R}^{(j)} \mathbf{p}_i^{(0)} + \mathbf{t}^{(j)}, i = 1, \dots, n\}, \quad (3.2)$$

where $\mathbf{p}_i^{(0)} \in \mathcal{P}^{(0)}$.

2. Find the closest point set $\mathcal{Q}^{(j+1)}$ on the target mesh \mathcal{M}_t for the updated point set $\mathcal{P}^{(j+1)}$:

$$\mathcal{Q}^{(j+1)} = \{\mathbf{q}_i^{(j+1)} = \arg \min_{\mathbf{q} \in \mathcal{M}_t} d(\mathbf{p}_i^{(j+1)}, \mathbf{q}), i = 1, \dots, n\}, \quad (3.3)$$

where $\mathbf{p}_i^{(j+1)} \in \mathcal{P}^{(j+1)}$, $d(\cdot, \cdot)$ is a distance function. The search could be accelerated by the k -d tree searching (Bentley, 1975).

3. Compute the similarity transformation from the initial point set $\mathcal{P}^{(0)}$ to the closest point set $\mathcal{Q}^{(j+1)}$ and use it as the next estimation:

$$(s^{(j+1)}, \mathbf{R}^{(j+1)}, \mathbf{t}^{(j+1)}) = \arg \min_{s, \mathbf{R}, \mathbf{t}} \sum_{i=1}^n \|s \mathbf{R} \mathbf{p}_i^{(0)} + \mathbf{t} - \mathbf{q}_i^{(j+1)}\|^2, \quad (3.4)$$

where $\mathbf{p}_i^{(0)} \in \mathcal{P}^{(0)}$, and $\mathbf{q}_i^{(j+1)} \in \mathcal{Q}^{(j+1)}$. This transformation estimation problem has the closed-form solution (Horn, 1987; Horn et al., 1988).

4. Terminate the loop when the number of iterations reaches the maximum number of iterations N , or the change in root-mean-square error falls below a specified threshold $\tau > 0$, i.e., $|e^{(j)} - e^{(j+1)}| < \tau$. The mean-square error for the j -th iteration is

$$e^{(j)} = \sqrt{\frac{1}{n} \sum_{i=1}^n \|s^{(j)} \mathbf{R}^{(j)} \mathbf{p}_i^{(0)} + \mathbf{t}^{(j)} - \mathbf{q}_i^{(j)}\|^2}, \quad (3.5)$$

where $\mathbf{p}_i^{(0)} \in \mathcal{P}^{(0)}$, and $\mathbf{q}_i^{(j)} \in \mathcal{Q}^{(j)}$.

The initial value of the similarity transformation $(s^{(0)}, \mathbf{R}^{(0)}, \mathbf{t}^{(0)})$ is important for the success of the registration. The similarity transformation is estimated based on a priori assumptions about the range scan that the main part of the range scan is a frontal face and the orientation of the range

scan is roughly known. However, in Section 3.9, I will demonstrate that this registration algorithm is robust with regard to initial values.

The initial scale s_0 is set to be the ratio of the average radii of the source and target triangle meshes. Let us denote the vertex positions of the source and the target meshes by $\{\mathbf{x}_{s,i}, i = 1, \dots, n_s\}$ and $\{\mathbf{x}_{t,i}, i = 1, \dots, n_t\}$, respectively. Then we have

$$s^{(0)} = \frac{\sum_{i=1}^{n_t} \|\mathbf{x}_{t,i} - \bar{\mathbf{x}}_t\| / n_t}{\sum_{i=1}^{n_s} \|\mathbf{x}_{s,i} - \bar{\mathbf{x}}_s\| / n_s}, \quad (3.6)$$

where $\bar{\mathbf{x}}_s$ and $\bar{\mathbf{x}}_t$ are the centroids of the source and target meshes. Assuming that the template and the range scan roughly face in the same direction, the initial rotation $\mathbf{R}^{(0)}$ is set to the identity matrix, *i.e.*,

$$\mathbf{R}^{(0)} = \mathbf{I}, \quad (3.7)$$

where \mathbf{I} is the 3×3 identity matrix. The initial translation $\mathbf{t}^{(0)}$ is given by

$$\mathbf{t}^{(0)} = \bar{\mathbf{x}}_t - s^{(0)} \mathbf{R}^{(0)} \bar{\mathbf{x}}_s. \quad (3.8)$$

3.5 Deformable Model

To emulate a plausible smooth surface deformation, this section starts from a physically accurate formulation of surface deformation, successively simplifies the computational model in order to achieve high efficiency and increased numerical stability, and finally presents a deformable model for surface deformation.

Let us consider a source surface \mathcal{S} to be deformed to a target surface \mathcal{T} (See Figure 3.4). From the physical point of view, there are two kinds of forces acting on the source surface \mathcal{S} : on the one hand, there are attractive forces that pull the source surface \mathcal{S} to the target surface \mathcal{T} ; on the other hand, there are internal forces that resist such deformation of the source surface \mathcal{S} such that the shape does not change dramatically. The two kinds of forces eventually result in that the surface \mathcal{S} becomes \mathcal{S}' in the equilibrium state.

Therefore, a energy function E_t of the surface \mathcal{S} consists of two parts: a deformation energy E_d associated with the internal forces, which measures how much the surface has resisted the deformation from its initial configuration, and a potential energy E_p associated with the attractive forces, which

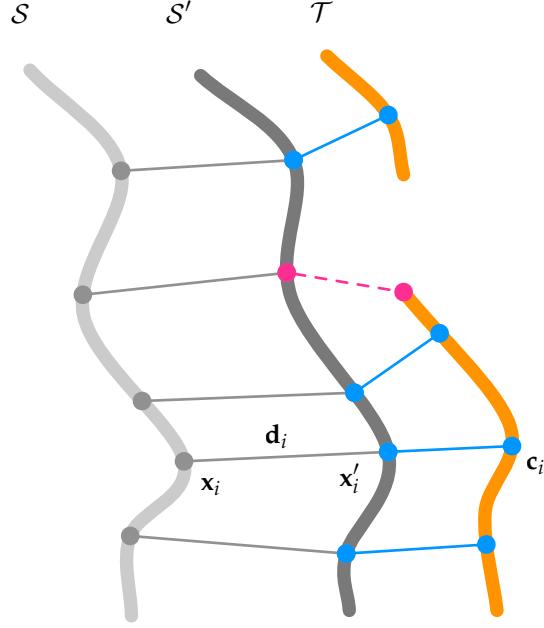


Figure 3.4: An illustration of surface deformation from \mathcal{S} to \mathcal{T} . The deformed surface \mathcal{S}' is represented by $\mathbf{x}'_i = \mathbf{x}_i + \mathbf{d}_i$, where $\mathbf{x}_i \in \mathcal{S}$ and $\mathbf{x}'_i \in \mathcal{S}'$. The point \mathbf{c}_i is the closest point to \mathbf{x}_i on the target surface \mathcal{T} . The dashed line illustrates the mismatching point to be disregarded.

measures how much the surface is driven towards the target surface \mathcal{T} , i.e.,

$$E_t = E_d + E_p. \quad (3.9)$$

The deformation energy E_d can be defined as the norm of the difference between the first and second fundamental forms of the surfaces \mathcal{S}' and \mathcal{S} (Terzopoulos et al., 1987),

$$\int_{\mathcal{S}} k_s l^2 \|\mathbf{I}' - \mathbf{I}\|_F^2 dA + \int_{\mathcal{S}} k_b l^4 \|\mathbf{II}' - \mathbf{II}\|_F^2 dA, \quad (3.10)$$

where $\mathbf{I} \in \mathbb{R}^{2 \times 2}$ and $\mathbf{II} \in \mathbb{R}^{2 \times 2}$ are the first and second fundamental forms of the surface \mathcal{S} , \mathbf{I}' and \mathbf{II}' are the first and second fundamental forms of the surface \mathcal{S}' , k_s and k_b are dimensionless parameters, l is the characteristic length of the surface \mathcal{S} , and $\|\cdot\|_F^2$ is the Forbenius matrix form. In the above equation, the first term represents the stretching energy and the second term represents the bending energy.

Since Equation 3.10 is nonlinear, it is generally simplified by replacing the changes of the first and second fundamental forms by the first-order and

second-order partial derivatives of the displacement function \mathbf{d} (Botsch and Sorkine, 2008),

$$E_d = \int_S k_s l^2 \left\| \frac{\partial \mathbf{d}}{\partial \mathbf{x}} \right\|_F^2 dA + \int_S k_b l^4 \left\| \frac{\partial^2 \mathbf{d}}{\partial \mathbf{x}^2} \right\|_F^2 dA, \quad (3.11)$$

where $\mathbf{d} = \mathbf{x}' - \mathbf{x}$, $\mathbf{x} \in S$, $\mathbf{x}' \in S'$, $\partial/\partial \mathbf{x}$ and $\partial^2/\partial \mathbf{x}^2$ are the first-order and the second-order partial derivatives with respect to the surface S .

The potential energy is calculated from the distance between the surfaces S' and \mathcal{T} . Given a mapping function $\mathbf{c} : S \mapsto \mathcal{T}$, we can write the potential energy can as

$$E_p = \int_S \frac{1}{2} w \|\mathbf{x}' - \mathbf{c}\|^2 dA = \int_S \frac{1}{2} w \|\mathbf{x} + \mathbf{d} - \mathbf{c}\|^2 dA, \quad (3.12)$$

where $w = w(\mathbf{x})$ is a weighting function, and $\|\cdot\|$ is the vector L^2 norm.

In the equilibrium state, the attractive forces and the internal forces balance that the shape of the surface S' does not change as well as the total energy. The first-order variational derivative of the energy equals to zero, *i.e.*,

$$\frac{\delta E_t}{\delta \mathbf{d}} = \frac{\delta E_d}{\delta \mathbf{d}} + \frac{\delta E_p}{\delta \mathbf{d}} = 0. \quad (3.13)$$

The first-order variational derivative of E_d can be written as

$$\frac{\delta E_d}{\delta \mathbf{d}} = \int_S \left(-k_s l^2 \Delta_{\mathbf{x}} \mathbf{d} + k_b l^4 \Delta_{\mathbf{x}}^2 \mathbf{d} \right) dA, \quad (3.14)$$

where $\Delta_{\mathbf{x}}$ is the Laplace-Beltrami operator with respect to the surface S (See Appendix B.5).

The first-order variational derivative of E_p is

$$\frac{\delta E_p}{\delta \mathbf{d}} = \int_S w(\mathbf{x} + \mathbf{d} - \mathbf{c}) dA. \quad (3.15)$$

Substituting Equations 3.14 and 3.15 into Equation 3.13 and using the fundamental lemma of the calculus of variations, we obtain the governing function

$$-k_s l^2 \Delta_{\mathbf{x}} \mathbf{d} + k_b l^4 \Delta_{\mathbf{x}}^2 \mathbf{d} + w(\mathbf{x} + \mathbf{d} - \mathbf{c}) = 0. \quad (3.16)$$

Thus, the deformed surface S' can be derived by adding the corresponding displacements solved from the above equation to the points of the original source surface S .

In Equation 3.16, the characteristic length l makes the parameters k_s and k_b scale-invariant, *i.e.*, independent of the scale of the surface \mathcal{S} . It can be any quantity measuring the characteristic size of the surface, such as the size of the bounding box and the average radius. When the surface \mathcal{S} is properly scaled in order to match with a target surface \mathcal{T} , its characteristic length is also scaled accordingly. However, the parameters k_s and k_b can remain unchanged. In the thesis, the characteristic length l is set to the size of the bounding box of the surface \mathcal{S} .

Discretisation To discretise Equation 3.16, let us consider a triangle mesh \mathcal{M} with n points $\{\mathbf{x}_i, i = 1, \dots, n\}$. Accordingly, the continuous displacement field \mathbf{d} and the mapping \mathbf{c} are replaced by the discrete counterparts, $\{\mathbf{d}_i, i = 1, \dots, n\}$ and $\{\mathbf{c}_i, i = 1, \dots, n\}$. With the discrete Laplace-Beltrami operator \mathcal{L} (See Section B.5), Equation 3.16 can be expressed in the matrix form

$$(-k_s l^2 \mathcal{L} + k_b l^4 \mathcal{L}^2 + \mathbf{W})\mathbf{D} = \mathbf{W}\mathbf{R}, \quad (3.17)$$

where \mathbf{W} is a $n \times n$ diagonal matrix with $W_{ii} = w_i$, \mathbf{D} is a $n \times 3$ matrix with the i -th row being the transpose of the unknown displacement vector \mathbf{d}_i , and \mathbf{R} is a $n \times 3$ matrix with the i -th row being $(\mathbf{c}_i - \mathbf{x}_i)^T$. The matrix on the left side is a sparse matrix. Figure 3.5 shows a visualisation of the sparse matrix, which shows that most non-zero elements concentrate along the diagonal line of the matrix. The sparse linear system is solved by using the UMFPACK numeric package (UMFPACK).

Control parameters The parameters k_s and k_b control the influence of deformation on the source surface. The deformable model produces the following different kinds of deformations through the combination of the parameters:

- For $k_s = 0$ and $k_b = 0$, there are no constraints on the smoothness of the surface, thus the closest points become the final points of the surface.
- For $k_s = 0$ and $k_b > 0$, the model only has the bending constraint. The deformation of the surface is similar to that of thin-plates.
- For $k_s > 0$ and $k_b = 0$, the model only has the stretching constraint. The deformation of the surface is similar to that of membranes.

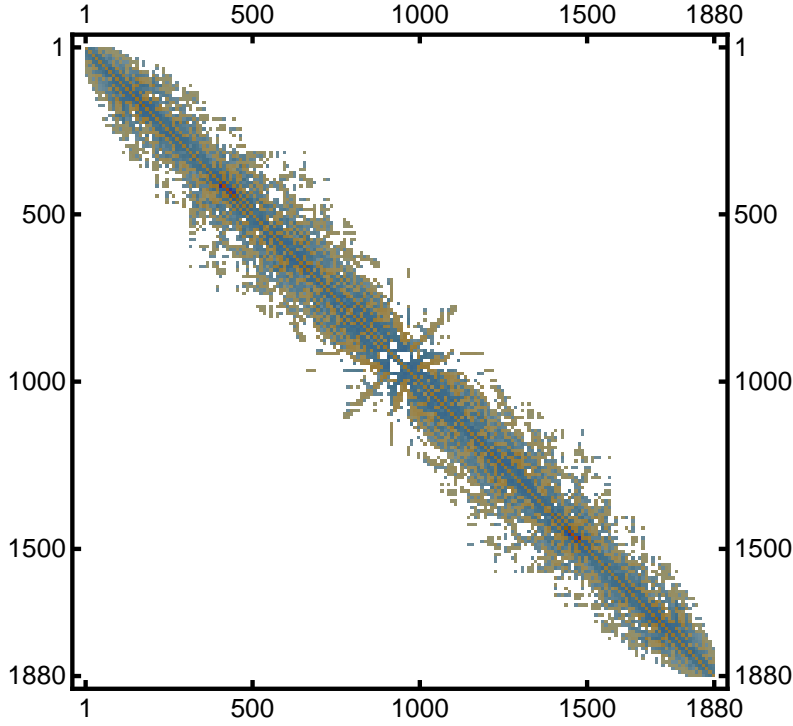


Figure 3.5: A visual plot of the sparse matrix of the deformable model. Only non-zeros are shown in dots.

- For $k_s > 0$ and $k_b > 0$, the model has both the stretching and bending constraints. The deformation of the surface is similar to that of thin-shells.

If k_s and k_b progressively decrease to zero, then the surface \mathcal{S} gradually relaxes its stiffness, becomes more and more flexible, and is eventually deformed onto those corresponding closest points.

Correspondences In order to deform the source mesh into the target mesh, it is important to establish good correspondences between them. In this thesis, the correspondences are built through an iterative process, which solves the same linear system of Equation 3.17 decreasing k_s and k_b each time and updating the closest points after each iteration. Each time the problem is solved, the template is deformed from its original undeformed state. Since k_s and k_b decrease, the template mesh more closely approximates the shape of the target after each iteration.

Weights The weighting function in Equation 3.12 shows the confidence of the correspondences. For its simplicity, a binary weighting function is used in this thesis,

$$w_i = \begin{cases} 0, & \text{if } \|\mathbf{c}_i - \mathbf{x}_i\| \leq d_m; \\ 1, & \text{if } \|\mathbf{c}_i - \mathbf{x}_i\| > d_m. \end{cases} \quad (3.18)$$

where d_m is a threshold value, which is set to the value that cut off the top 10% among the distances between the corresponding points

3.6 Surface Reconstruction

This section describes the steps of using a template to reconstruct the surface of a range scan by using the registration approach and the deformable model proposed in Sections 3.4 and 3.5. In the following text, let us refer to the template and the range scan as the source mesh \mathcal{M}_s and the target mesh \mathcal{M}_t .

1. Roughly estimate the initial scale, rotation, and translation and globally align the source mesh \mathcal{M}_s with the target mesh \mathcal{M}_t (See Section 3.4 for more details).
2. Repeatedly deform the aligned source mesh \mathcal{M}_s into the target mesh \mathcal{M}_t by using the deformable model.
 - a) Initially, set $\mathbf{x}_i^{(0)} \in \mathcal{M}_s$, $k_s^{(0)} = k_b^{(0)} = 100$.
 - b) Compute the discrete Laplace-Beltrami operator \mathcal{L} with respect to the source mesh \mathcal{M}_s (See Section B.5).
 - c) Find the closest points $\mathbf{c}_i^{(j+1)} \in \mathcal{M}_t$ to the points $\mathbf{x}_i^{(j)}$.
 - d) Compute the weights $w_i^{(j+1)}$.
 - e) Solve Equation 3.17 and obtain the displacements \mathbf{d}_i^{j+1} .
 - f) Update the points: $\mathbf{x}_i^{(j+1)} = \mathbf{x}_i^{(0)} + \mathbf{d}_i^{(j+1)}$.
 - g) Decrease the parameters: $k_s^{(j+1)} = k_s^{(j)} / 2$, $k_b^{(j+1)} = k_b^{(j)} / 2$.
 - h) Go to the step (c) if $j \leq N$. Otherwise, terminate the iteration. The final deformed source mesh is given by the points $\mathbf{x}_i^{(j+1)}$.

3.7 Texture Reconstruction

Most range scans have realistic photograph textures associated with them. It would be desirable to make the textures available for reconstructed facial models as well. Since noisy range scans contain holes and missing data, the texture coordinates is incomplete. This section shows how to use the deformable model to recover the texture coordinates for the reconstructed facial models.

In Section 3.5, the displacement field was considered as the unknown function defined on the source surface. The displacement field was reconstructed from the displacements estimated from correspondences while satisfying the smoothness constraints. This method can be extended to recover the texture coordinates, that is, consider the texture coordinates as the unknown function and reconstruct them from the texture coordinates that are associated with the corresponding points on the target surface. Let us denote the texture coordinates by $\mathbf{u} \in \mathbb{R}^2$. Then Equation 3.16 can be rewritten as

$$-k_s l^2 \Delta_{\mathbf{x}} \mathbf{u} + k_b l^4 \Delta_{\mathbf{x}}^2 \mathbf{u} + w(\mathbf{u} - \mathbf{v}) = 0, \quad (3.19)$$

where \mathbf{v} is the corresponding texture coordinates on the target surface. Similarly, the discrete version of the equation is

$$(-k_s l^2 \mathcal{L} + k_b l^4 \mathcal{L}^2 + \mathbf{W})\mathbf{U} = \mathbf{WV}, \quad (3.20)$$

It is worth to note that Equations 3.17 and 3.20 have the same sparse matrix. Thus, it can be factorised once and reused. Actually, to reconstruct the textures, only one back-substitution step needs to be added into the last iteration in Section 3.6.

3.8 Facial Expression Tracking

This section presents the procedure of facial expression tracking for 4D range scans in a frame-to-frame manner. Given a sequences of range scans of a human face performing facial expressions, $\mathcal{S} = \{\mathcal{M}_{\text{scan}}^{(t)}, t = 0, \dots, n\}$, without loss of generality, let us denote by $\mathcal{M}_{\text{scan}}^{(0)}$ the reference neutral range scan. The template facial model $\mathcal{M}_{\text{template}}$ is first registered to $\mathcal{M}_{\text{scan}}^{(0)}$ using the proposed registration method and the aligned template model $\bar{\mathcal{M}}_{\text{template}}^{(0)}$ is obtained. Then the proposed deformable model is used to deform the

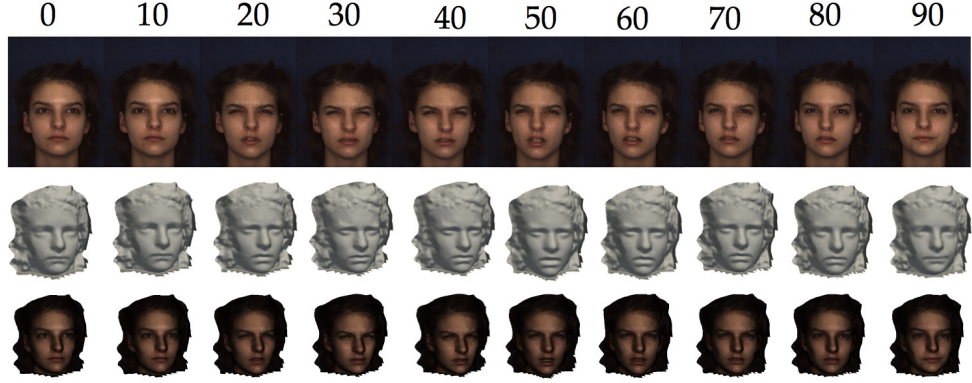


Figure 3.6: A sequence of range scans in the FED database. The 4D range scans are the sequence of a female’s angry expression. The first row shows the 2D texture images. The second and third rows show the range scans rendered without and with textures. The frame numbers are listed at the top of the figure.

initial aligned template $\tilde{\mathcal{M}}_{\text{template}}^{(0)}$ to $\mathcal{M}_{\text{scan}}^{(0)}$, yielding the reconstructed facial model $\mathcal{M}_{\text{template}}^{(0)}$ for that range scan. For the subsequent range scans, we deform the previous deformed template $\mathcal{M}_{\text{template}}^{(t)}$ to $\mathcal{M}_{\text{scan}}^{(t+1)}$. This tracking procedure can be described in Equation 3.21.

$$\begin{array}{ccc}
 \mathcal{M}_{\text{scan}}^{(t)} & & \mathcal{M}_{\text{scan}}^{(t+1)} \\
 \downarrow & & \downarrow \\
 \mathcal{M}_{\text{template}}^{(t)} & \longrightarrow & \mathcal{M}_{\text{template}}^{(t+1)}
 \end{array} \tag{3.21}$$

Dataset The dataset for the tracking is from the high-resolution 3D dynamic facial expression database (FED) Yin et al. (2008), which has been used for face information analysis. The FED database collected the spatio-temporal range scan sequences of six facial expressions (angry, disgust, fear, happy, sad, surprise) for each person. The sequences of range scans are captured at a speed of 25 frames per second. Each facial expression last about 4 seconds, thus there are about 100 frames of range scans for each expression. For each frame of the sequence, there are a 2D texture image and a 3D range scan. Figure 3.6 shows an example of 4D range scans in the FED database.

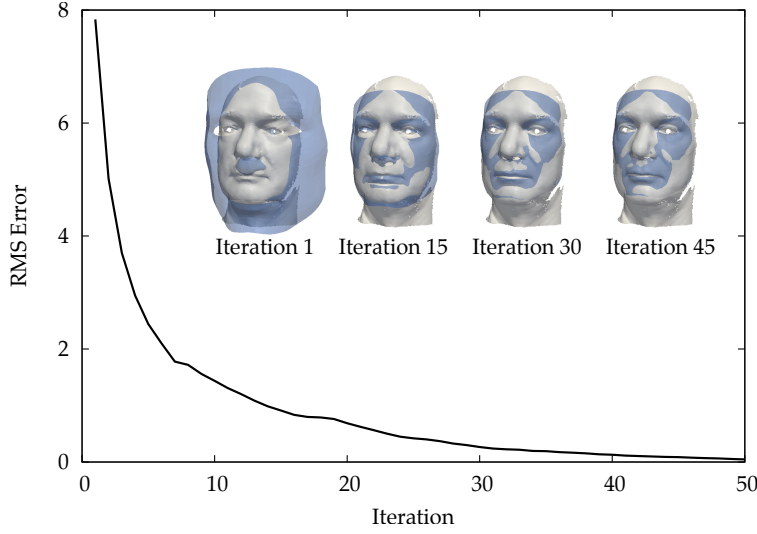


Figure 3.7: The RMS error curve of a registration. The declining curve of the RMS errors indicates the convergence of the registration. The inset figures show the overlaps of the template and the range scan at four different iterations.

3.9 Results and Discussion

When the template was registered to a range scan by using the registration method presented in Section 3.4, the RMS error $e^{(j)}$ of each iteration measured the overall distance between the range scan and the template that is transformed by the current estimated transformation. Thus, the convergence of a declining curve of the RMS errors indicated that the transformed template was gradually matched with the range scan, for example, the orientation and the scale between the template and the range scan were matched. Figure 3.7 shows the curve of the RMS errors of a registration. From the figure, we can see that the template is quickly matched with the range scan in the scale and the orientation.

In order to know how initial scales affect the convergence of a registration, several tests were carried out with different initial scales. The initial scales were obtained through multiplying the reference scale computed from Equation 3.6 by the factors listed in Figure 3.8. The curves of the RMS errors were plotted as logarithmic scale in Figure 3.8 in order to distinguish the speed of convergence. The solid line in Figure 3.8 shows the path of

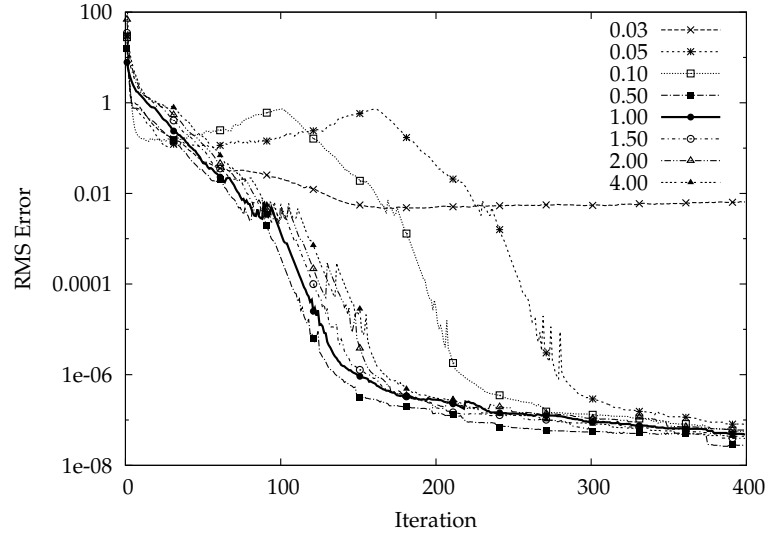
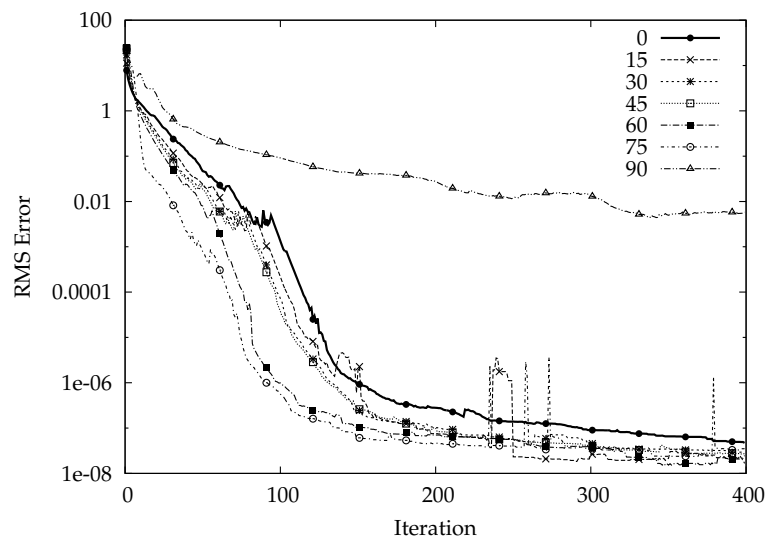
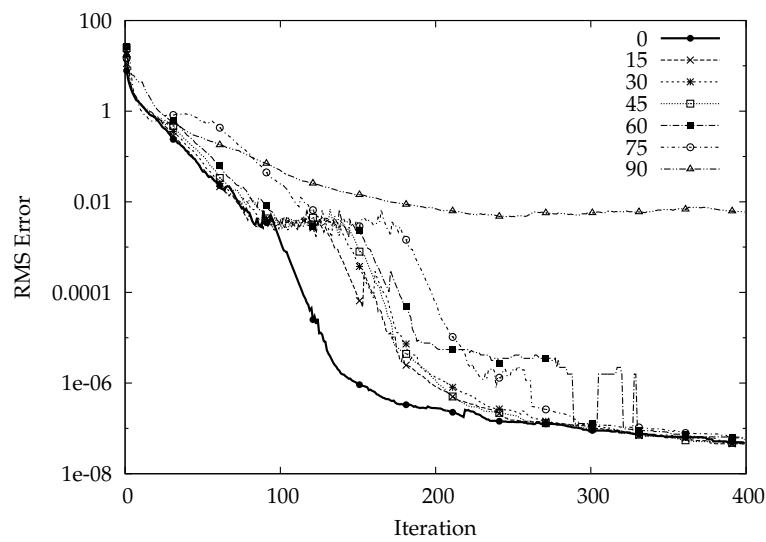
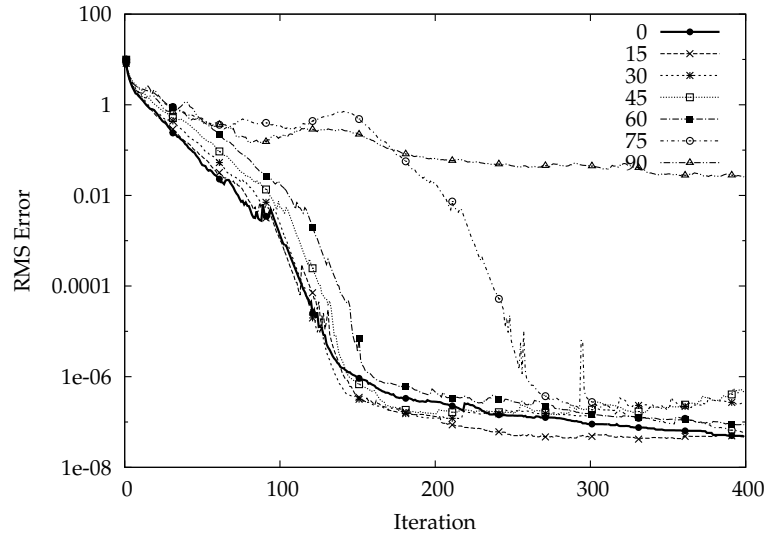


Figure 3.8: The influence of initial scales on the convergence of a registration. The initial scales of the template are obtained through multiplying the reference scale computed from Equation 3.6 by the factors listed in the figure. Among all the initial scales, the registration converges except for the factor of 0.03. The vertical axis is in logarithmic scale.

a successful registration. For differential initial scales, if the curve of the RMS errors eventually converges to the solid line, then the registration is also successful. From Figure 3.8, we can see that the registration quickly converges for the initial scales obtained from the factors in a wide range of 0.5–4.0. For the initial scales being about only 5% and 10% of the reference scale, the registration can still converge after 300 iterations. However, if the initial scale is too small comparing to that of the range scan, the registration may fail to converge due to that the template may be trapped in local minimisations. For example, for the initial scale being only 3% of the reference scale, the registration is unsuccessful due to that the template is trapped in a small portion of the range scan. For this case, the curve of the RMS errors will not coincide with the solid line. These results demonstrate that the registration algorithm is robust with regard to the initial scale estimated from Equation 3.6.

Figure 3.9 shows the influence of initial relative rotations on the convergence of a registration. Initially, both the template and the range scan face towards the direction of the z-axis and their upright directions are along the

(a) Initial rotations about the x -axis(b) Initial rotations about the y -axis



(c) Initial rotations about the z-axis

Figure 3.9: The influence of the initial relative rotations about (a) the x -axis, (b) the y -axis, and (c) the z -axis on the convergence of a registration. Initially, both the template and the range scan face towards the direction of the z -axis and their upright directions are along the y -axis. Then, eighteen relative rotations with respect to the x , y , and z -axes, listed in degrees in the figures (a), (b) and (c), are applied to the template. The tests are carried out between the transformed templates and the range scan. Among all the tests, the registration converges except for 90 degrees. The vertical axes are in logarithmic scale.

y -axis. Then, the rotations of 15, 30, 45, 60, 75, 90 degrees, about the x , y , and z -axes, respectively, are applied to the template. The tests are carried out between the transformed templates and the range scan. The results are shown in Figures 3.9 (a), (b), and (c), respectively. Similarly, the curves are plot as logarithmic scale in order to distinguish the speed of convergence and the solid line denotes the successful registration without changing the initial rotation. From these figures, we can see that the registration converges for all initial relative rotations less than 75 degrees.

After the registration, the template was further deformed towards the range scan. To demonstrate such a deformation process, Figure 3.10 shows a close-up of the nose region of the template with the range scan shown as semi-transparent surfaces for comparison. This figure clearly shows that the template is smoothly deformed towards the range scan in 10 iterations. The

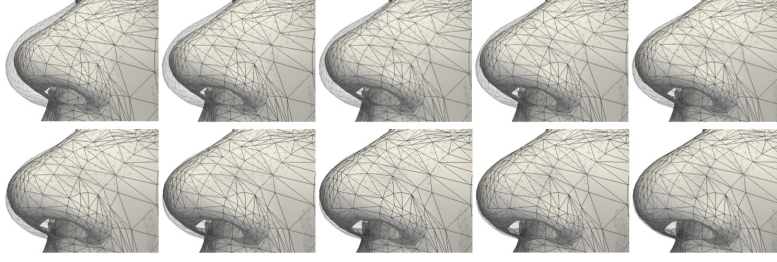


Figure 3.10: A close-up in the nose region during the iterative deformation by using the deformable model. Parameters: $k_s^0 = k_b^0 = 100$, $N = 10$. The range scan is shown as semi-transparent meshes, which can be easily seen in the first few figures.

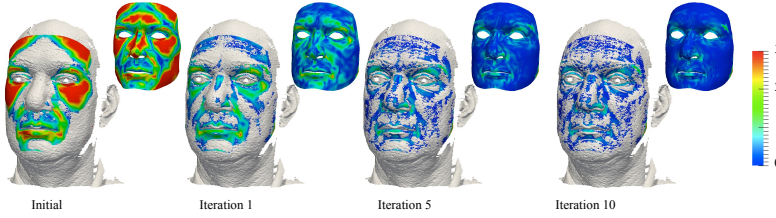


Figure 3.11: The deformation process of the template towards the range scan by using the deformable model. Parameters: $k_s^0 = k_b^0 = 100$, $N = 10$. The colour mapping shows the distances between the template and the range scan.

final deformed template captures the shape of the range scan and keeps the topological structures of the original template.

Figure 3.11 shows the deformation process during the surface reconstruction of a range scan. The distances from the template to the range scan are encoded into colours with the red colour indicating large distances and the blue colour indicating small distances. Both the colour mapping and the overlaps clearly show that the template is smoothly deformed onto the range scan.

Figure 3.12 shows the result of the texture reconstruction. The range scan and the reconstructed facial model are rendered with the checkboard texture and the original texture, respectively. From the figure, we can see that the facial features are faithfully matched in the template and the range scan. The reconstructed facial model with the reconstructed texture is more realistic than the original range scan as the holes are filled and the noise level is reduced.



Figure 3.12: The result of the texture reconstruction. On the left side, the checkboard texture is used for comparison; on the right side, the facial image texture is used.

Figure 3.13 shows more results of the facial model reconstruction of the range scans in the FRGC v2.0 dataset. The first and the second columns are the range scans shown in shaded and texture-mapped renderings. The third (fourth) column shows the overlaps between the range scans (grey) and the rigid (non-rigid) registered template models (blue). The final reconstructed facial models are shown in the last two columns in shaded and texture-mapped renderings. All these reconstructed facial models have the same mesh structures. As shown in the figure, even for range scans containing hair, our method can still find the correct facial regions and reconstruct the facial models.

Figures 3.14 and 3.15 show the results of facial expression tracking. The first column is the range scans and the template facial model, the second column is the neutral faces, and the rest columns are six facial expressions (angry, disgust, fear, happy, sad, surprise), which are presented in the range scan sequences. The corresponding deformed template facial models are shown in the second row. The results show that the proposed method is able to track the facial expressions expected for that with very large deformation (See the last expression in Figure 3.15).

There are some limitations of the facial expression tracking method. Firstly, current tracking method only employed the closest point constraints, which would not be able to track large and fast motion, especially in the chin region (see the surprise expression in Figure 3.15). The motion of the chin often exhibits fast and abrupt and hence the deformable model can fail to track it correctly. Secondly, the method was based on the assumption that the change of facial motion between two consecutive frames was small. Thirdly, the texture information was not used in the model.



Figure 3.13: The results of automatic 3D facial model and texture reconstruction. The range scans, shown in shaded and texture-mapped renderings in the first and the second columns, are from the FRGC v2.0 dataset. The third (fourth) column shows the overlaps between the range scan (grey) and the rigid (non-rigid) registered template model (blue). The final reconstructed facial models are shown in the last two columns in shaded and texture-mapped renderings. All these reconstructed models have the same mesh structures.

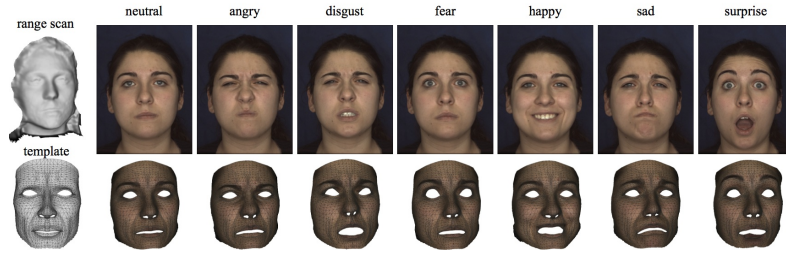


Figure 3.14: The results of facial expression tracking (female).

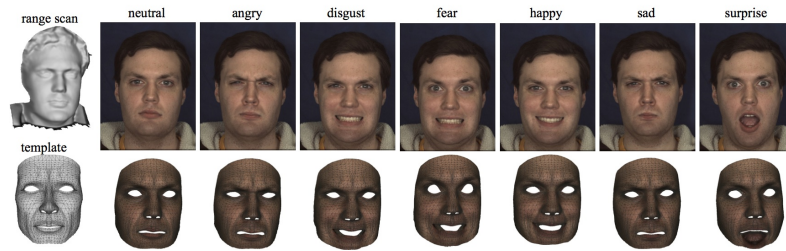


Figure 3.15: The results of facial expression tracking (male).

3.10 Chapter Summary

In this chapter, an automatic approach for 3D facial model and texture reconstruction from range scans has been developed. This approach is fully automatic and does not require any manual interactions or landmarks.

The registration method is able to automatically adjust the scale and the orientation of the template in order to match it with a noisy range scan. The results show that the registration method is robust with regard to initial scales and rotations.

The proposed deformable model is capable of smoothly deforming the aligned template towards the range scan. Therefore, the final deformed template recovers the shape of the range scan as well as the texture. Due to hole filling and smoothing of the noisy data, the reconstructed results are more realistic than the original range scans. The reconstructed facial models have the same mesh structures, which could provide the dense correspondences across the facial models to further analysis of the facial models.



Facial Expression Transferring

This chapter presents an automatic approach to transferring facial expressions from one facial model to another and the results has been partially published in the paper (Xiang et al., 2009). This approach encodes existing facial expressions on a facial model in the form of displacement deformation gradients and reconstructs similar facial expressions on a new facial model by building the corresponding guidance deformation gradients for the new model. The displacement deformation gradients of a facial expression are computed by using the discrete gradient operator (Appendix B) on the motion vectors of the expression; and the deformable model developed in Chapter 3 is used to establish the correspondences of triangles in order to build guidance deformation gradients from the computed displacement deformation gradients. The representation of encoding facial expressions in displacement deformation gradients is scale-invariant. Thus, the approach is capable of transferring facial expressions between facial models with different sizes. Furthermore, our approach minimises the local errors of deformation gradients due to shape differences and miss-matching correspondences across the entire facial model in the least-squares sense and produces plausible results.

The automatic facial expression transferring approach greatly reduces the work load to animators and artists for facial animation. When creating similar facial animations for different virtual characters in a film production or a computer game, animators/artists need to create the key frames

of various facial expressions for only one facial model and then they can use the automatic approach to clone the existing facial expressions to other facial models instead of creating similar facial animations from scratch. This would enable a character to be replaced with a new character and maintain the similar expression in virtual reality.

In the following, the approach to automatically transferring facial expressions will be elaborated in more detail. Firstly, Section 4.1 presents an overview of the problem of facial expression transferring and the challenges of achieving the facial expression transferring. Then, Section 4.2 describes the representation of facial expressions. Section 4.3 presents the details of encoding facial expressions in displacement deformation gradients. Section 4.4 presents the details of deformation reconstruction from given displacement deformation gradients. Section 4.5 show how to build guidance deformation gradients for facial expression transferring. Finally, the results and the conclusion of this chapter are presented in Sections 4.6 and 4.7.

4.1 Overview

A facial expression involves groups of facial muscle movements. These movements convey the emotional state of the individual to observers. However, in this thesis, only the shapes of faces are considered and thus facial expressions are manifested by the geometrical changes of the shapes of faces.

To measure quantitatively a facial expression, a reference face and a proper deformation need to be defined metrically to measure the deformation from the reference face to the expression face. It is natural to choose a neutral face as the reference since the neutral face relaxes all facial muscles and thus has no expressions. Thus, a facial expression can be represented through its neutral face with a proper deformation metric.

Since different individuals have different neutral facial shapes, the question is how a facial expression can be transferred from one person to another such that the transferred expression still looks similar to the original one. To be precise, we give the following problem statement (see Figure 4.1):

Given a source facial model S with a neutral facial shape S_0 and an expression facial shape S_1 , and a target facial model T with a neutral facial shape T_0 , transfer the facial expression $E(S_0, S_1)$ presented in the expression facial shape S_1 to the target model T such that the transferred expression $E'(T_0, T_1)$ looks similar to the

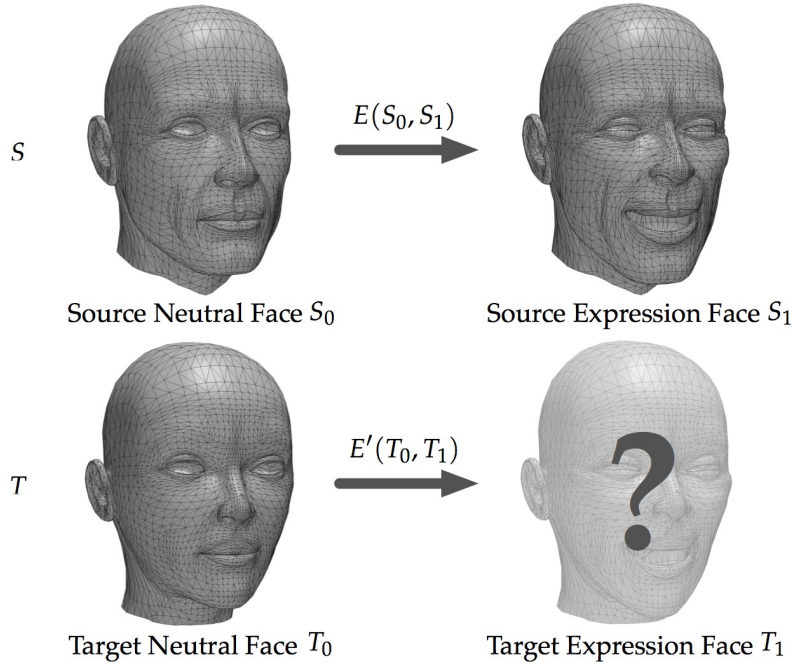


Figure 4.1: The problem of facial expression transferring: Transfer the facial expression $E(S_0, S_1)$ from a source model S to a target model T such that the transferred facial expression $E'(T_0, T_1)$ is similar to $E(S_0, S_1)$ as much as possible.

original one $E(S_0, S_1)$ as much as possible.

The above problem poses two challenges. The requirement of the target model having the similar expression as that of the source means that the mouth lips of the target should deform like that of the source, the mouth and eye corners of the target like that of the source, and so on. This implies that we must bring the two different models, the source and the target, into correspondences with each other. Section 4.5.1 presents the details of establishing the correspondences by using the deformable model developed in Section 3.5.

Since different facial models might have different sizes and orientations, it is obvious that we cannot directly apply the displacement field from the source to the target. Figure 4.4 shows a few examples of the issues with directly transferring the expression displacements to the same face subject to the uniform scale and rotation transformations. From the figure, we can see that the method of directly transferring unmodified displacements poorly

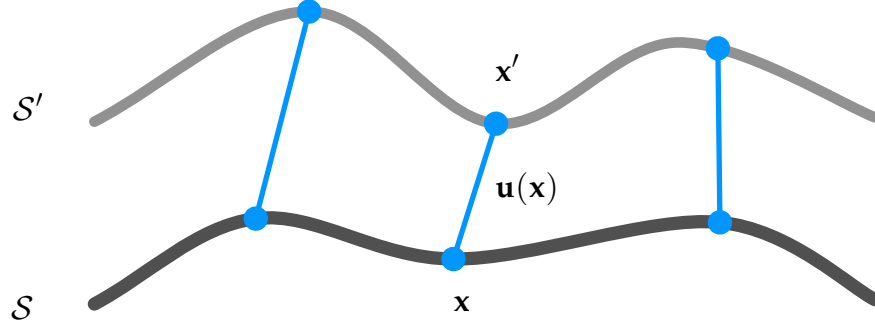


Figure 4.2: An illustration of the displacement representation of facial expressions.

replicated an expression on targets, and even resulted in artefacts. To solve this problem, Noh and Neumann (2001) proposed a complex heuristic algorithm to locally adapt the directions and the scales of the displacements on a new model. Sumner and Popović (2004) used the affine transformations of corresponding triangles as the deformation metric for transferring arbitrary deformation between triangle meshes. In this thesis, the problem of facial expression transferring is formulated within the framework of differential geometry and use the displacement deformation gradients as the deformation metric to represent facial expressions.

4.2 Facial Expression Representation

As mentioned above, a facial expression of a face can be considered as the deformation of its expression face with respect to its neutral face. Let us consider a neutral face \mathcal{S} and its expression face \mathcal{S}' (Figure 4.2). Let \mathbf{x} denote the position of a point on \mathcal{S} and \mathbf{x}' the corresponding point on \mathcal{S}' . One of the simplest approaches to describing the deformation of the facial expression is the displacement representation, which measures the displacement vector field between the two facial models, *i.e.*,

$$\mathbf{u} = \mathbf{x}' - \mathbf{x}, \quad (4.1)$$

where $\mathbf{x} \in \mathcal{S}$, $\mathbf{x}' \in \mathcal{S}'$. Figure 4.3 shows an example of the displacement representation of a laughing expression, where the displacement vectors of the laughing expression are visualised by using colour mapping and arrows.

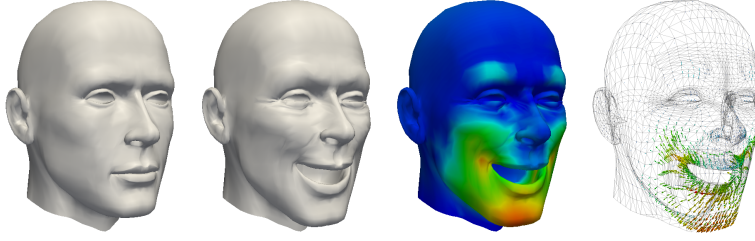


Figure 4.3: The displacement representation of a facial expression. The first and second images show the neutral and expression faces, respectively. The displacement vectors of the expression are visualised by using colour mapping and arrows in the third and fourth images.

Although the displacement representation is simple, it has some limitations on facial expression transferring. Firstly, the displacement representation is scale-dependent. It means that, if we want to transfer a facial expression to a face with a different scale, we need to properly adjust the scale the displacement vectors in order to produce similar deformation on that face; otherwise, artefacts would occur. For example, the second and third images in Figure 4.4 show the artefacts due to directly applying the expression displacements to the same face but having different scale transformations. Secondly, the displacement representation is rotation-dependent because the directions of displacement vectors are dependent on rotations. The last image of Figure 4.4 shows the artefacts due to directly applying expression displacements on the same face rotated by 60 degrees.

In this thesis, the displacement deformation gradients are considered as the representation of facial expressions. The deformation gradient representation is scale-invariant and can be easily transferred onto different faces. The next section presents the details of this representation.

4.3 Displacement Deformation Gradients

A facial expression can be represented as a displacement function defined on the neutral face \mathcal{S} (Figure 4.2), *i.e.*,

$$\mathbf{u} = \mathbf{u}(\mathbf{x}), \quad (4.2)$$

where \mathbf{u} is the displacement function and $\mathbf{x} \in \mathcal{S}$. Now, let us study how the displacement function changes with respect to the shape of the neutral

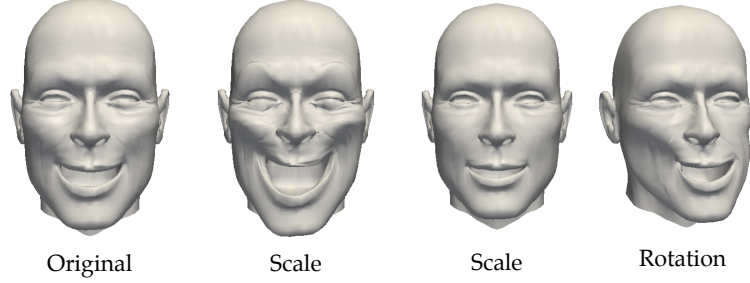


Figure 4.4: Issues with directly transferring expression displacements. The first figure is the original face with a laughing expression and the reset figures are the faces that are obtained by directly transferring the displacements of the expression onto the same neutral face subject to different transformations. The transformations (from left to right) are the uniform scale by 0.5 and 2, and the rotation by 60 degrees.

face. Assuming the displacement function $\mathbf{u}(\mathbf{x})$ is continuously differentiable with respect to \mathbf{x} , we can express the infinitesimal change $d\mathbf{u}$ at \mathbf{x} as

$$d\mathbf{u} = \frac{\partial \mathbf{u}}{\partial \mathbf{x}} \cdot d\mathbf{x} = \nabla_{\mathbf{x}} \mathbf{u} \cdot d\mathbf{x} = \mathbf{F} \cdot d\mathbf{x}, \quad (4.3)$$

where $\nabla_{\mathbf{x}}$ denotes the gradient operation with respect to the \mathbf{x} coordinates and \mathbf{F} is called the displacement deformation gradient tensor defined as

$$\mathbf{F} = \frac{\partial \mathbf{u}}{\partial \mathbf{x}}. \quad (4.4)$$

The deformation gradient tensor \mathbf{F} measures the local change of the displacement function of a facial expression with respect to the shape of the neutral face. As we shall see in the next section, once the displacement deformation gradients of a facial expression are known, we can easily reconstruct the corresponding displacements of the facial expression. Thus, the displacement deformation gradients can also be used as a deformation metric to encode facial expressions.

Now, let us generalise the above displacement deformation gradient from continuous settings to discrete settings. Consider a neutral face and an expression face represented by the triangle meshes $\mathcal{M} = (\mathcal{K}, \mathcal{P})$ and $\mathcal{M}' = (\mathcal{K}, \mathcal{P}')$, respectively, where the topological component \mathcal{K} contains n vertices $\{v_i, i = 1, \dots, n\}$ and m triangles $\{t_j, j = 1, \dots, m\}$, the vertex posi-

The figure shows two triangles. The left triangle, labeled 'Undeformed triangle $t_j \in \mathcal{K}$ ', has vertices \mathbf{x}_{j1} , \mathbf{x}_{j2} , and \mathbf{x}_{j3} . The right triangle, labeled 'Deformed triangle $t_j \in \mathcal{K}$ ', has vertices \mathbf{x}'_{j1} , \mathbf{x}'_{j2} , and \mathbf{x}'_{j3} . Below these diagrams is the linear system of discrete displacement deformation gradients:

$$\mathbf{F}_j = (\nabla_{\mathbf{x}}\phi_{j1}(\mathbf{x}) \nabla_{\mathbf{x}}\phi_{j2}(\mathbf{x}) \nabla_{\mathbf{x}}\phi_{j3}(\mathbf{x})) \begin{pmatrix} \mathbf{u}_{j1}^T \\ \mathbf{u}_{j2}^T \\ \mathbf{u}_{j3}^T \end{pmatrix}$$

$$\begin{pmatrix} \mathbf{F}_1 \\ \vdots \\ \mathbf{F}_j \\ \vdots \\ \mathbf{F}_m \end{pmatrix}_{3m \times 3} = \begin{pmatrix} t_1 \\ \vdots \\ t_j \\ \vdots \\ t_m \end{pmatrix} \begin{pmatrix} v_1 & \cdots & v_{j1} & \cdots & v_{j2} & \cdots & v_{j3} & \cdots & v_n \\ & \ddots & & & & & & & \\ & & \nabla_{\mathbf{x}}\phi_{j1}(\mathbf{x}) & \cdots & \nabla_{\mathbf{x}}\phi_{j2}(\mathbf{x}) & \cdots & \nabla_{\mathbf{x}}\phi_{j3}(\mathbf{x}) & & \\ & & & & & & & \ddots & \end{pmatrix}_{3m \times n} \begin{pmatrix} \mathbf{u}_1^T \\ \vdots \\ \mathbf{u}_{j1}^T \\ \vdots \\ \mathbf{u}_{j2}^T \\ \vdots \\ \mathbf{u}_{j3}^T \\ \vdots \\ \mathbf{u}_n^T \end{pmatrix}_{n \times 3}$$

$\mathbf{F}_{3m \times 3} \qquad \mathbf{G}_{3m \times n} \qquad (\mathbf{u}^T)_{n \times 3}$

Figure 4.5: The linear system of discrete displacement deformation gradients, where \mathbf{F} is a $3m \times 3$ matrix, representing the discrete deformation gradients; \mathbf{G} a $3m \times n$ matrix, representing the discrete gradient operator of the undeformed mesh (neutral face) \mathcal{M} ; \mathbf{u}^T a $n \times 3$ matrix, representing the vertex displacements of the deformed mesh (expression face) \mathcal{M}' with respect to \mathcal{M} ; $\phi_{jk}(\mathbf{x}), k = 1, 2, 3$ the linear basis functions of the triangle t_j ; \mathbf{F}_j the deformation gradient of the triangle t_j .

tion of the neutral face $\mathcal{P} = \{\mathbf{x}_i, i = 1, \dots, n\}$ and the vertex positions of the expression face $\mathcal{P}' = \{\mathbf{x}'_i, i = 1, \dots, n\}$.

Since the topological structure of the mesh is assumed to remain unchanged during the deformation, each vertex v_i of \mathcal{M} changes its position from \mathbf{x}_i to \mathbf{x}'_i . Thus, the vertex displacements can be written as

$$\mathbf{u}_i = \mathbf{x}'_i - \mathbf{x}_i. \quad (4.5)$$

According to the definition of discrete potential fields in Appendix B, the set of vertex displacements $\{\mathbf{x}_i, i = 1, \dots, n\}$ forms a discrete vector potential field of \mathcal{M} . Using the discrete gradient operator obtained in Section B.3, we can write the gradients of the vertex displacements in the matrix form

$$\mathbf{F} = \mathbf{G}\mathbf{u}^T, \quad (4.6)$$

where the superscript T denotes the transpose of a matrix, \mathcal{G} the discrete gradient operator, \mathbf{F} the discrete gradient matrix, and \mathbf{u}^T the assembled matrix of the vertex displacement (See Figure 4.5). The discrete gradient operator \mathcal{G} only depends on the topological and geometrical structures of the mesh \mathcal{M} . Section B.3 shows how to build the matrix in more detail.

Due to the piecewise linearity of triangle meshes, for each triangle $t_j \in \mathcal{K}$, the deformation gradient \mathbf{F}_j is constant and the column vectors of \mathbf{F}_j are coplanar with the triangle. The set of these constant gradients, denoted by

$$\mathbf{F} = \{\mathbf{F}_j, j = 1, \dots, m\}, \quad (4.7)$$

forms the discrete displacement deformation gradients of the expression face \mathcal{M}' with respect to the neutral face \mathbf{M} .

Figure 4.6 shows a few examples of the displacement deformation gradients for a triangle subject to some simple deformations, such as in-plane and off-plane rotations, shearing, and scaling. The x , y , and z -component vectors of the gradients are visualised in the arrows with the red, green, and blue colours. All these component vectors remain constant within the triangles and are coplanar with the planes of the undeformed triangles.

4.4 Deformation Reconstruction

The previous section has shown how to represent a facial expression in terms of discrete displacement deformation gradients on a triangle mesh. In this section, the inverse problem will be addressed, *i.e.*, how to reconstruct the discrete deformation displacements for given discrete deformation gradients on a triangle mesh.

Consider a triangle mesh \mathcal{M} with n vertices and m triangles. Given discrete deformation gradients $\{\mathbf{G}_i \in \mathbb{R}^{3 \times 3}, i = 1, \dots, m\}$ defined on \mathcal{M} with each gradient \mathbf{G}_i corresponding to the i -th triangle of \mathcal{M} , the goal is to find the displacements $\{\mathbf{u}_i, i = 1, \dots, n\}$ such that the gradients of the displacements matches the given guidance gradients as closely as possible. Mathematically, this problem can be expressed in a least-squares sense, *i.e.*,

$$\min_{\mathbf{u}} \|\mathcal{G}\mathbf{u}^T - \mathbf{G}\|_F^2, \quad (4.8)$$

where $\|\cdot\|_F$ is the Frobenius norm of a matrix, \mathcal{G} is a $3m \times n$ sparse matrix representing the discrete gradient operator of \mathcal{M} , \mathbf{u}^T is an $n \times 3$ matrix with

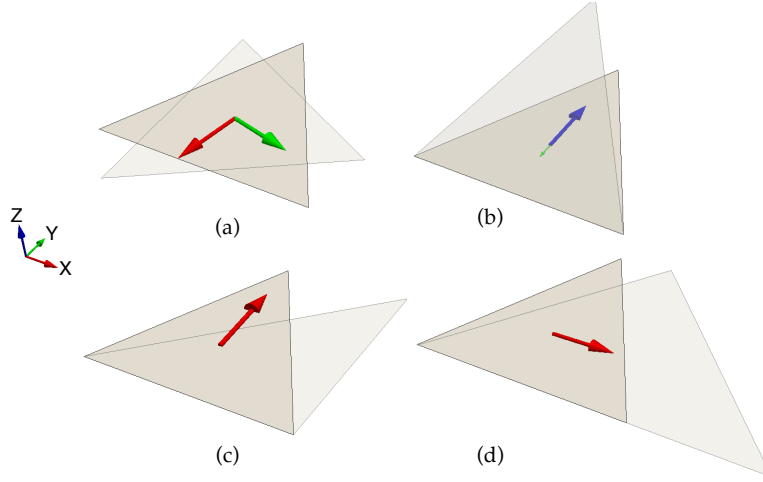


Figure 4.6: Examples of the displacement deformation gradients of a triangle subject to simple deformations: (a) in-plane rotation by 30 degrees; (b) off-plane rotation by 30 degrees; (c) shearing along the x -axis; (d) scaling along the x -axis by 2. The deformed configurations of the triangles are indicated in light-grey colours. The directions and magnitudes of the x , y , and z -components of the displacement gradients are shown in the arrows with red, green and blue colours, respectively. All these components are coplanar with the planes of the undeformed triangles.

the i -th row being \mathbf{u}_i^T , and \mathbf{G} is a $3m \times 3$ matrix with the i -th 3×3 matrix block being \mathbf{G}_i . The term $\mathcal{G}\mathbf{u}^T$ represents the gradient of the unknown displacement \mathbf{u} . However, since gradients are invariant under translations, we need to specify the displacement for one vertex of \mathcal{M} in order to obtain a unique solution to the minimisation problem. Without loss of generality, let us set $\mathbf{u}_0 = \mathbf{0}$.

Expanding the matrix norm in Equation 4.8 gives

$$\begin{aligned} \|\mathcal{G}\mathbf{u}^T - \mathbf{G}\|_F^2 &= \text{tr}((\mathcal{G}\mathbf{u}^T - \mathbf{G})^T(\mathcal{G}\mathbf{u}^T - \mathbf{G})) \\ &= \text{tr}(\mathbf{u}\mathcal{G}^T\mathcal{G}\mathbf{u}^T) - 2\text{tr}(\mathbf{G}^T\mathcal{G}\mathbf{u}^T) + \text{tr}(\mathbf{G}^T\mathbf{G}), \end{aligned} \quad (4.9)$$

where $\text{tr}(\cdot)$ is the trace of a matrix.

According to vector and matrix calculus (Petersen and Pedersen, 2008), taking the derivative of this expression with respect to \mathbf{u} yields

$$\frac{d\|\mathcal{G}\mathbf{u}^T - \mathbf{G}\|_F^2}{d\mathbf{u}} = 2\mathbf{u}\mathcal{G}^T\mathcal{G} - 2\mathbf{G}^T\mathcal{G}. \quad (4.10)$$

Finally, setting the derivative equal to zero gives the equation

$$\mathbf{u}\mathcal{G}^T\mathcal{G} = \mathbf{G}^T\mathcal{G}, \quad (4.11)$$

or

$$\mathcal{G}^T\mathcal{G}\mathbf{u}^T = \mathcal{G}^T\mathbf{G}, \quad (4.12)$$

where $\mathcal{G}^T\mathcal{G}$ is an $n \times n$ sparse symmetric matrix, $\mathcal{G}^T\mathbf{G}$ is an $n \times 3$ matrix, and \mathbf{u}^T is an $n \times 3$ matrix containing unknown displacements. This sparse linear system can be efficiently solved using a sparse matrix solver such as UMFPACK (UMFPACK), and then the deformed configuration of \mathcal{M} can be reconstructed by adding the obtained displacements to the vertex positions of \mathcal{M} .

It can be easily verified that, if the given guidance gradients are exactly the same as the displacement deformation gradients, *i.e.*, $\mathbf{G} = \mathcal{G}\mathbf{u}^T$, Equation 4.8 attains the minimal value 0 and the reconstructed deformation only differs by a constant translation. Generally, the given gradients are not exactly the same as $\mathcal{G}\mathbf{u}^T$. Thus, the results are obtained in the least-squares sense.

4.5 Guidance Deformation Gradients

From the above section, we know that the motion vectors of a facial expression on a triangle mesh can be reconstructed in the least-squares sense from the deformation gradients specified on each triangle of the mesh. Thus, to transfer a facial expression from a source mesh to a target mesh is to build the corresponding deformation gradients, called guidance deformation gradients, on the target mesh from its deformation gradients on the source mesh. To do that, first we need to establish triangle correspondences between the two meshes, and then locally adjust deformation gradients from the source mesh to the target mesh according to the correspondences.

4.5.1 Correspondences

The correspondences between two meshes have to be established in order to transform deformation gradients from one mesh to another. Let us consider two triangle meshes $\mathcal{M}_s = (\mathcal{K}_s, \mathcal{P}_s)$ and $\mathcal{M}_t = (\mathcal{K}_t, \mathcal{P}_t)$, where the subscripts s and t denote the source and the target meshes. Since the deformation gradients are constant within each triangle, we only need to establish

the correspondences between triangles. Thus, we can define the correspondences between \mathcal{M}_s and \mathcal{M}_t as follows

$$\{(t_{s,j}, t_{t,k}) \mid t_{s,j} \in \mathcal{K}_s, t_{t,k} \in \mathcal{K}_t\}, \quad (4.13)$$

where $t_{s,j}$ and $t_{t,k}$ are triangles of \mathcal{M}_s and \mathcal{M}_t , respectively.

To establish the correspondences, the deformable model, developed in Chapter 3, is employed. It is possible to build the correspondences between \mathcal{M}_s and \mathcal{M}_t either by deforming \mathcal{M}_s into \mathcal{M}_t , or by deforming \mathcal{M}_t into \mathcal{M}_s . In both cases, the following discussion is almost the same. Thus, only the latter case is considered here.

The idea is that, using the deformable model, the mesh \mathcal{M}_t is deformed into $\widehat{\mathcal{M}}_t = (\mathcal{K}_t, \widehat{\mathcal{P}}_t)$, which has the shape of \mathcal{M}_s , but remains the same topological component as that of \mathcal{M}_t . Then the correspondences can be established from the two meshes $\widehat{\mathcal{M}}_t$ and \mathcal{M}_s by finding closest corresponding triangles with each other. However, some triangles of $\widehat{\mathcal{M}}_t$ might not have corresponding triangles on \mathcal{M}_s and vice versa. Thus, to avoid mismatching, we need to limit the search range under the threshold value

$$d_m = c \cdot \max(\bar{l}_s, \bar{l}_t), \quad (4.14)$$

where c is a factor, \bar{l}_s and \bar{l}_t are the average edge length of the meshes \mathcal{M}_s and \mathcal{M}_t , respectively. In this thesis, $c = 3$.

To be precise, for a triangle $t_{s,j} \in \mathcal{K}_s$, we find the closest triangle $t_{t,k} \in \mathcal{K}_t$ with the distance $d(t_{s,j}, t_{t,k}) \leq d_m$ and then the two triangles form a correspondence, *i.e.*,

$$\mathcal{C}_s = \bigcup_{t_{s,j} \in \mathcal{K}_s} \{(t_{s,j}, t_{t,k}) \mid d(t_{s,j}, t_{t,k}) = \min, d(t_{s,j}, t_{t,k}) \leq d_m, t_{t,k} \in \mathcal{K}_t\}, \quad (4.15)$$

where $d(\cdot, \cdot)$ is a distance function, which is chosen as the distance between the centroids of two triangles in $\widehat{\mathcal{M}}_t$ and \mathcal{M}_s . To make the correspondences symmetric, we also consider the correspondences from \mathcal{M}_t to \mathcal{M}_s , that is,

$$\mathcal{C}_t = \bigcup_{t_{t,j} \in \mathcal{K}_t} \{(t_{s,j}, t_{t,k}) \mid d(t_{s,j}, t_{t,k}) = \min, d(t_{s,j}, t_{t,k}) \leq d_m, t_{s,k} \in \mathcal{K}_s\}. \quad (4.16)$$

Then the correspondences between \mathcal{M}_s and \mathcal{M}_t is

$$\mathcal{C} = \mathcal{C}_s \cup \mathcal{C}_t. \quad (4.17)$$

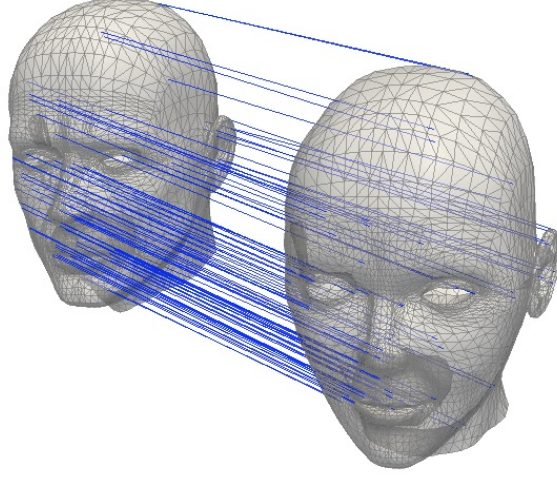


Figure 4.7: A visual illustration of the correspondences established between two facial models. Each line connecting two triangles represents a correspondence. In this figure, only 100 lines of sampled correspondences are shown in order to make the figure clear.

It should be noted that, for a triangle in \mathcal{M}_s , there might exist many triangles in \mathcal{M}_t corresponding to it, and vice versa. Thus, the set \mathcal{C} is a many-to-many relationship.

Figure 4.7 visually shows an example of the correspondences between two facial models established by using the deformable model that is developed in Chapter 3. The lines that connect corresponding triangles indicate the correspondences of the two models. In the figure, only a small number of correspondences are visible in order to make the lines distinguishable from each other. This figure shows that all facial features of the two models are well matched.

4.5.2 Local Transformation

Once the triangle correspondences \mathcal{C} are established between \mathcal{M}_s and \mathcal{M}_t , the deformation gradients \mathcal{F}_s on \mathcal{M}_s can be locally adjusted to \mathcal{M}_t so that the adjusted gradients \mathcal{F}_t can be used as guidance deformation gradients to reconstruct the facial expression. The guidance deformation gradients field \mathcal{F}_t can be built in two steps (Figure 4.8): first, project \mathcal{F}_s from \mathcal{M}_s onto $\widehat{\mathcal{M}}_t$ and get $\widehat{\mathcal{F}}_t$; then locally transform $\widehat{\mathcal{F}}_t$ from $\widehat{\mathcal{M}}_t$ to \mathcal{M}_t and obtain \mathcal{F}_t .

Consider a triangle $t_{t,k} \in \mathcal{K}_t$. Let us denote the set of triangles corre-

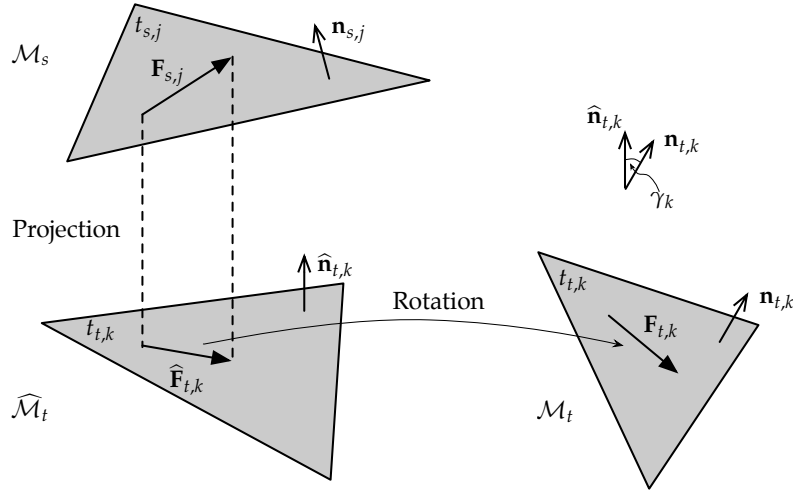


Figure 4.8: An illustration of the local transformation for deformation gradients.

sponding to $t_{t,k}$ on \mathcal{M}_s by $\mathcal{C}(t_{t,k})$, i.e.,

$$\mathcal{C}(t_{t,k}) = \{t_{s,j} \mid (t_{s,j}, t_{t,k}) \in \mathcal{C}, t_{s,j} \in \mathcal{K}_s\}. \quad (4.18)$$

If $\mathcal{C}(t_{t,k}) = \emptyset$, that is, there are no corresponding triangles to $t_{s,k}$, then let $\widehat{\mathbf{F}}_{t,k} = \mathbf{0}$; otherwise, for each corresponding triangle $t_{s,j} \in \mathcal{C}(t_{t,k})$, project its gradient matrix $\mathbf{F}_{s,j}$ onto the triangle $t_{t,k}$, and then average all the projected gradients, i.e.,

$$\widehat{\mathbf{F}}_{t,k} = \frac{1}{|\mathcal{C}(t_{t,k})|} \sum_{t_{s,j} \in \mathcal{C}(t_{t,k})} (\mathbf{I} - \widehat{\mathbf{n}}_{t,k} \widehat{\mathbf{n}}_{t,k}^T) \mathbf{F}_{s,j}, \quad (4.19)$$

where $|\cdot|$ denotes the cardinal number of a set, \mathbf{I} is the identity matrix, and $\widehat{\mathbf{n}}_{t,k}$ is the unit normal of the triangle $t_{t,k}$.

Next, the projected gradients $\widehat{\mathbf{F}}_{t,k}$ on $\widehat{\mathcal{M}}_t$ need to be rotated so that they become coplanar with the triangles of \mathcal{M}_t . In Figure 4.8, the angle of the corresponding triangles of \mathcal{M}_t and $\widehat{\mathcal{M}}_t$ is

$$\gamma_k = \cos^{-1} \mathbf{n}_{t,k} \cdot \widehat{\mathbf{n}}_{t,k}, \quad (4.20)$$

where $\mathbf{n}_{t,k}$ and $\widehat{\mathbf{n}}_{t,k}$ are the unit normal vectors to the corresponding triangles of \mathcal{M}_t and $\widehat{\mathcal{M}}_t$, respectively. The rotation can then be expressed in the quaternion (Weisstein)

$$q_k = \left(\cos \frac{\gamma_k}{2}, (\widehat{\mathbf{n}}_{t,k} \times \mathbf{n}_{t,k})^T \sin \frac{\gamma_k}{2} \right). \quad (4.21)$$

Table 4.1: The topology information of the facial models in Figure 4.9

| Model Name | Number of Vertices | Number Triangles |
|------------|--------------------|------------------|
| Template | 3861 | 7678 |
| Boy | 2674 | 5304 |
| Girl | 1773 | 3516 |
| Judy | 3861 | 7678 |
| Penny | 3861 | 7678 |

Finally, the guidance deformation gradient $\mathbf{F}_{t,k}$ for the triangle $t_{t,k}$ can be written as

$$\mathbf{F}_{t,k} = \mathbf{R}_k \hat{\mathbf{F}}_{t,k}, \quad (4.22)$$

where \mathbf{R}_k is a 3×3 rotation matrix, which can be obtained from the quaternion q_k ¹.

4.6 Results and Discussion

Figure 4.9 shows the results of transferring facial expressions from a template facial model onto other four different facial models. These facial models and the expressions on the template model are extracted from Poser². The template model has six different facial expressions (impressed, laughing, pleased, rage, sad, surprise). These expression faces have the same mesh structures as the template. Using the proposed facial expression transferring method, these facial expressions are transferred onto four different facial models. Table 4.1 lists the topological information about these facial models in the same order.

Figure 4.10 shows the results of transferring five facial expressions from a template facial model onto three individual facial models, which are reconstructed from the range scans of the FRGC v2.0 dataset by using the facial modelling approach developed in Chapter 3.

¹Given a unit quaternion $q = (w, x, y, z)$, the equivalent post-multiplied 3×3 rotation matrix is (Bar-Itzhack, 2000)

$$\mathbf{R} = \begin{pmatrix} 1 - 2y^2 - 2z^2 & 2xy - 2zw & 2xz + 2yw \\ 2xy + 2zw & 1 - 2x^2 - 2z^2 & 2yz - 2xw \\ 2xz - 2yw & 2yz + 2xw & 1 - 2x^2 - 2y^2 \end{pmatrix}.$$

²<http://poser.smithmicro.com>

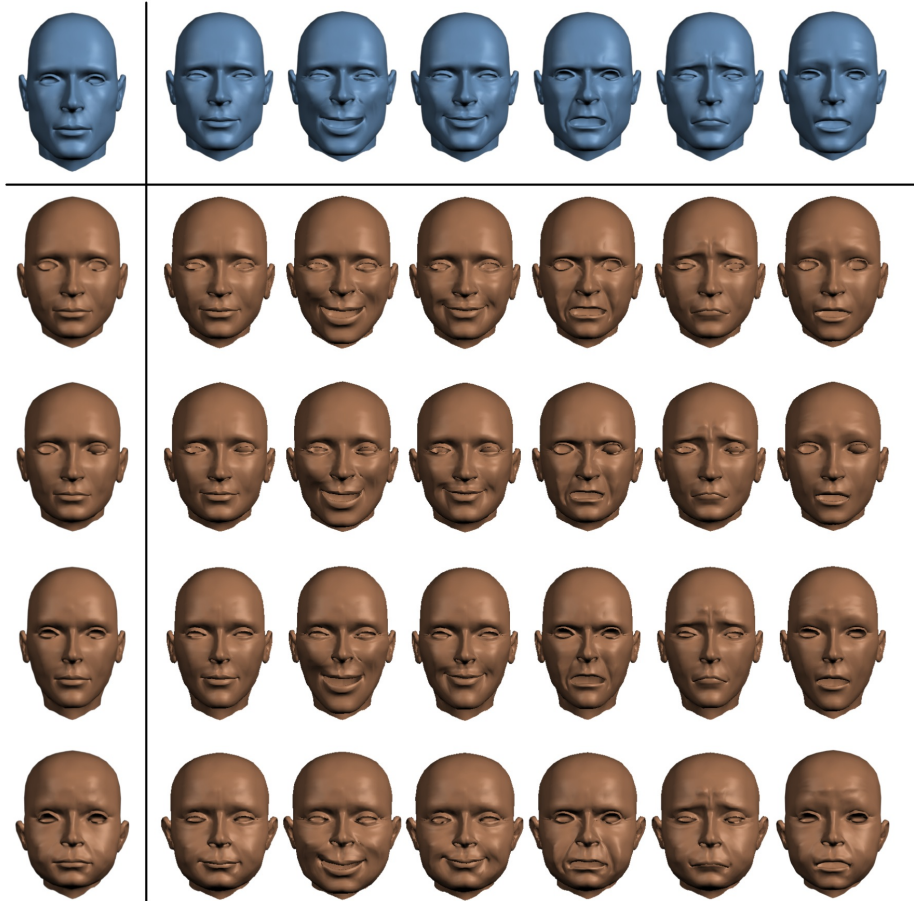


Figure 4.9: The results of transferring facial expressions from the template facial model onto other four different facial models. In the first row are the template model and six different facial expressions (impressed, laughing, pleased, rage, sad, surprise). The following rows are the individual models and their corresponding facial expressions transferred from the template.



Figure 4.10: The results of facial expression transferring on the reconstructed facial models from rang scans. Five facial expressions (anger, laughing, pleased, rage, sad) of the template facial model, shown in the first row, are transferred onto three different facial models, which are reconstructed from the range scans of the FRGC v2.0 dataset by using the facial modelling approach developed in Chapter 3. For each individual facial model, the results are rendered without and with textures.

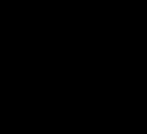
The time cost for transferring a facial expression onto a different facial model depends on the number of vertices of the two models. In the above examples, the whole process takes about 2–3 minutes for transferring a facial expression, timing results on 2.2GHz Intel Core 2 Duo computer.

From Figures 4.9 and 4.10, we can see that all the facial expressions on the template facial model are faithfully cloned onto other facial models. It is worth to note that the reconstructed facial models in Figure 4.10 have different sizes comparing to the template. This demonstrates that the representation of facial expressions proposed in this chapter is scale-invariant.

4.7 Chapter Summary

In this chapter, a new automatic approach to transferring facial expressions from one facial model to another has been proposed. This approach is developed within the framework of discrete differential geometry. It employs displacement deformation gradients instead of displacement vectors to encode facial expressions. This representation has the advantage of being scale-invariant. To transfer a facial expression from a source facial model to a target facial model, we need to build the corresponding guidance deformation gradients on the target facial model from the deformation gradients of the given expression, and then the facial expression is reconstructed on the target based on the guidance gradients in the least-squares sense.

This page is intentionally left blank.



Evaluation

In order to demonstrate the performance of the methods proposed in the previous two chapters, this chapter presents quantitative and qualitative evaluations of these methods. The rest of this chapter is organised as follows. The performance of the registration method and the comparison with other registration methods are shown in Section 5.1. Section 5.2 describes the evaluation of the deformable model. Next, Section 5.3 presents a qualitative evaluation of facial expression transferring. Finally, A summary of this chapter is given in Section 5.4.

5.1 Registration

Signed Distances When the template and the range scan are aligned with each other, the points of the template are likely to deviate in both directions from the the range scan. Thus, signed distances (Bærentzen and Aanæs, 2005) are used here to distinguish the distances from the template to the range scan in both sides.

RMS Error Given a set of signed distances $\{d_1, \dots, d_n\}$, the root-mean-square error e_{rms} can be written as

$$e_{rms} = \sqrt{\frac{\sum_{i=1}^n d_i^2}{n}} \quad (5.1)$$



Figure 5.1: Examples of the FRGC range scans containing outliers.

5.1.1 FRGC Dataset

The FRGC v2.0 database contains 4950 range scans in total. Most of the range scans have outliers such as hair, ears, and necks. Some range scans even have a considerable part of shoulders and clothes (Figure 5.1). In order to test the performance of the registration method developed in Chapter 3, two experiments were carried out on the FRGC v2.0 database.

Experiment 1 The dataset of the experiment contained 200 range scans, which were selected from the FRGC v2.0 database without removing any outlier range scans.

Experiment 2 One hundred range scans were first randomly selected from the database. Each range scan was manually checked and the outlier range scans with shoulders and necks were removed from the dataset. The final 50 range scans were used as the dataset of the experiment.

For each experiment, a generic template was registered to every range scan in the dataset with the maximum number of iterations set to 100.

The results of the two experiments are shown in Tables C.1 and C.2 of the Appendix C. For Experiment 1, 157 out of 200 registration tests are successful, and for Experiment 2, all of the registration tests are successful.

Figure 5.2 shows a successful registration in Experiment 1. The overlaps of the template and the range scan at different iteration steps are shown in the first and third rows. Below each overlap, there is a histogram of the signed distances from the template to the range scan. The signed distances are also visualised in colour mapping on the template. The range scan and the curve of the RMS error are given on the left side of the figure.

From Figure 5.2, we can see that the registration converges after 50 iterations. The scale and orientation of the template are gradually adjusted to match with the target range scan during the 50 iterations. The histogram

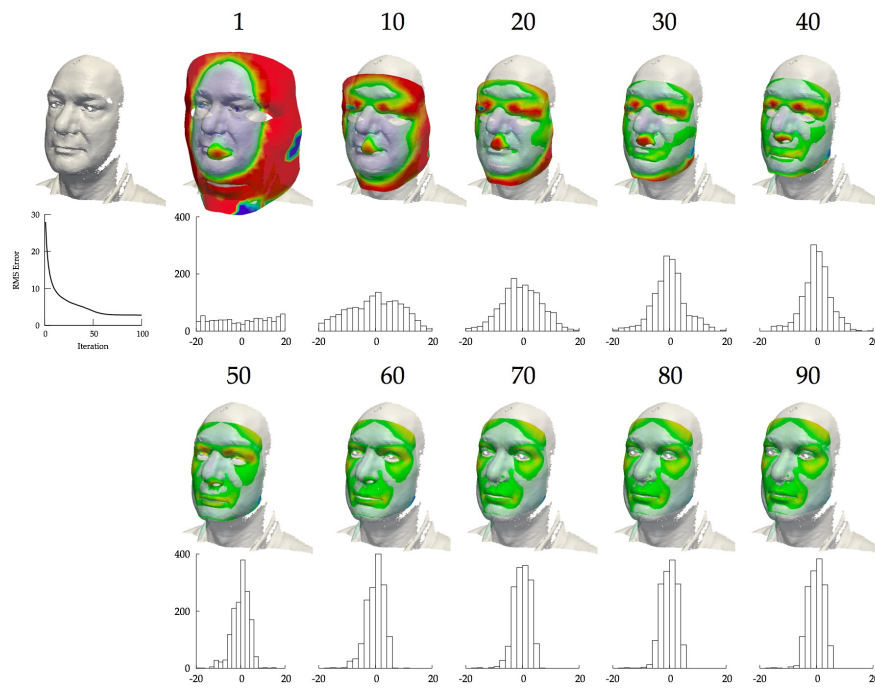


Figure 5.2: A successful registration for the FRGC dataset.

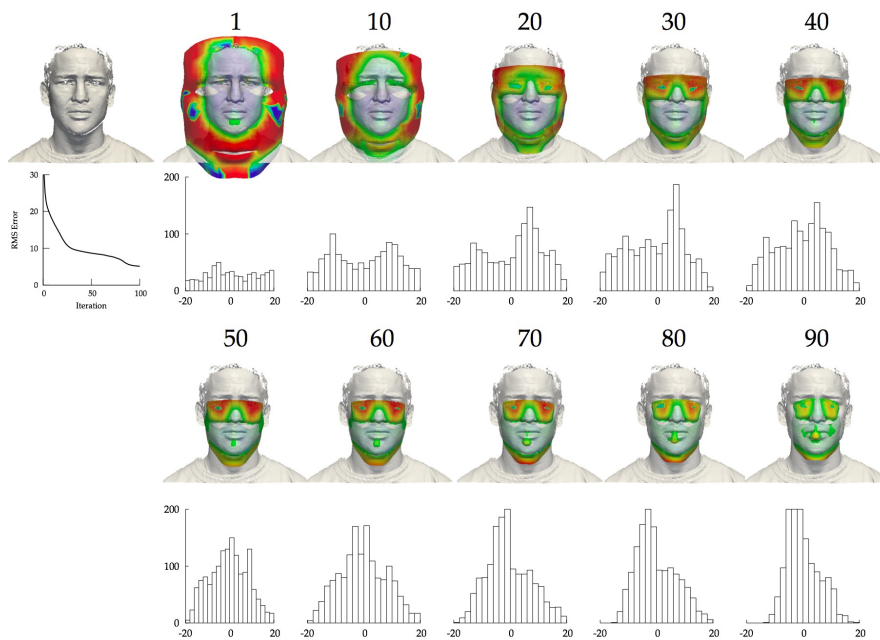


Figure 5.3: A failed registration for the FRGC dataset.

distribution of the signed distances gradually changes from a rather flat shape to a sharp peak shape centred at the zero axis. This also confirms that the alignment between the template and the range scan are refined step by step at the early stage of the registration.

A failed case of the registration for the FRGC dataset is shown in Figure 5.3. In this case, the facial region is only a small part of the range scan. The outliers of this range scan have a strong influence on the estimation of the initial transformations and correspondences.

The above results demonstrate that the registration method is able to align a template with a range scan by tuning the scale and orientation of the template and can handle certain outliers that are not major parts of the range scan.

5.1.2 Comparison

There are many surveys of registration methods which provided a comparison between already published approaches. For example, Rusinkiewicz and Levoy (2001) evaluated many variants of ICP on synthetically-generated test scenes using the baseline proposed by Pulli (1999). Salvi et al. (2007) compared registration methods for range scans, including Besl's method (Besl and McKay, 1992), Chen's method (Chen and Medioni, 1992), Zinsser's method (Zinsser et al., 2003), Trucco's method (Trucco et al., 1999), Chow's method (Chow et al., 2004) and the spin image method (Johnson and Hebert, 1999). These methods were evaluated on synthetic and real data. In order to compare the registration method of this thesis with other methods, the test data of the survey by Salvi et al. (2007) was used. The results are shown in Figures 5.4, 5.5 and 5.6.

5.2 Deformable Model

5.2.1 The Curve of the RMS Error

Similar to registration, the RMS error is still used to measure overall shape differences between a deforming template and a target. Figure 5.7 shows a curve of the RMS error for a local deformation process. Comparing to registration (Figure 3.7), the curve has a step shape with three stages:

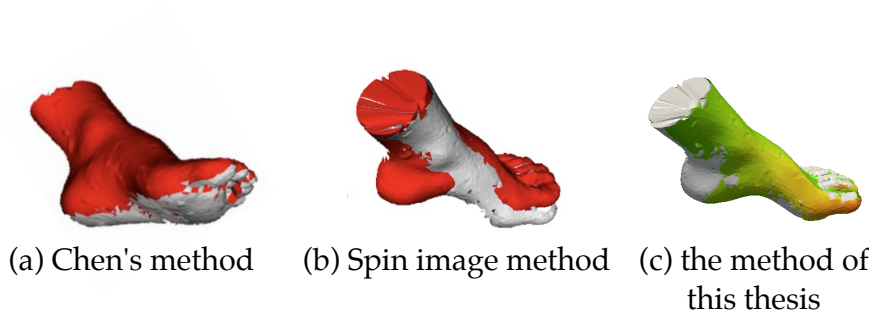


Figure 5.4: Registration of the synthetic data of a foot.

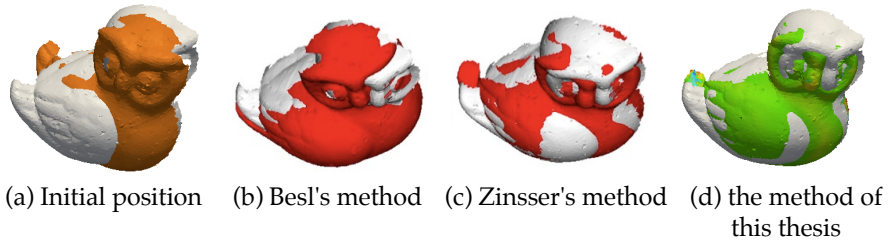


Figure 5.5: Registration of two range scans of an owl.

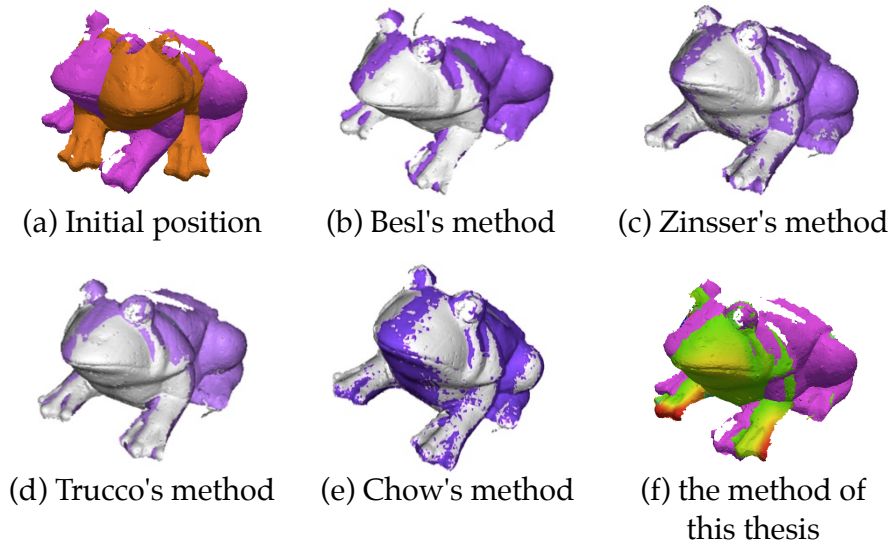


Figure 5.6: Registration of two range scans of a frog.

Stage 1 The initial rather flat portion of the curve is Stage 1. At this stage, the deformable model has large stiffness due to relative large values of the control parameters k_s and k_b . Thus, it is hard for the template to deform towards its target.

Stage 2 The declining portion of the curve is Stage 2. At Stage 2, the rigidity of the template is gradually reduced by decreasing the control parameters k_s and k_b . Thus, it becomes more and more flexible so that surface points of the template could be smoothly pulled towards the target.

Stage 3 Stage 3 is the rest portion of the curve. When the template is already deformed to the target, no more deformation will take place.

The slop of the curve at Stage 2 is related to the decreasing rate of the deformable model's control parameters. This curve can be used to determine the optimal values of the control parameters and the maximum number of iterations. For example, Figure 5.7 shows that the template does not deform further after 30 iterations, thus it would be better to set the maximum of iterations to 30 to reduce computational costs.

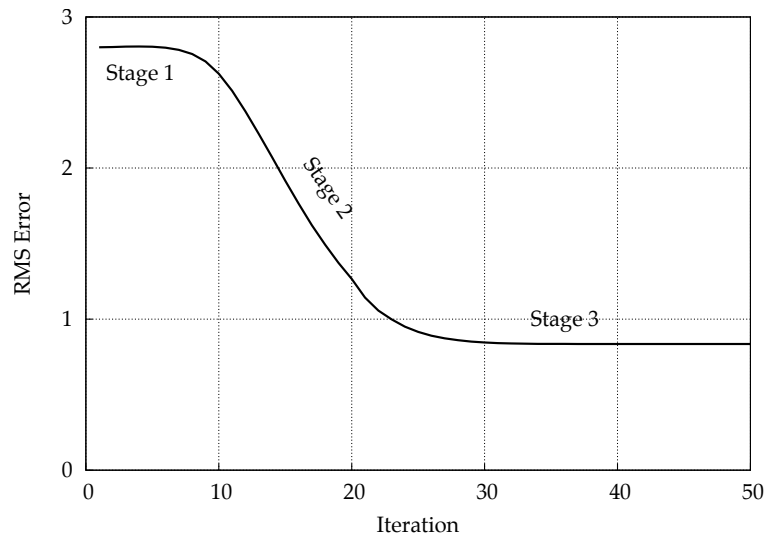


Figure 5.7: The curve of the RMS error for a deformation process.

5.2.2 Influence of the Control Parameters k_s and k_b

The control parameters k_s and k_b play an important role in the deformable model. In order to show their influence on deformation, two experiments were carried out by varying them independently. In the first experiment, while k_b was fixed at 0.01, k_s could vary from 0 to 1000. In the second experiment, k_s was set to 1.0 and k_b could change from 0 to 100. The results are shown in Figures 5.8 and 5.9. These two figures show that increasing either k_s or k_b results in right shifting of the RMS error curve, which means that the length of Stage 1 becomes longer.

5.2.3 Deformation

All the registered template in Section 5.1.1 were further deformed onto the target range scans using the deformable model (Refer to Tables C.1 and C.2 of the Appendix C for more details). Figure 5.10 shows an example deformation process. In the first row are overlaps of the deforming template and the target range scan at the indicated iterations. The colour mapping on the template visually shows the signed distances from the template to the target. The histograms of the distances are shown below each overlap.

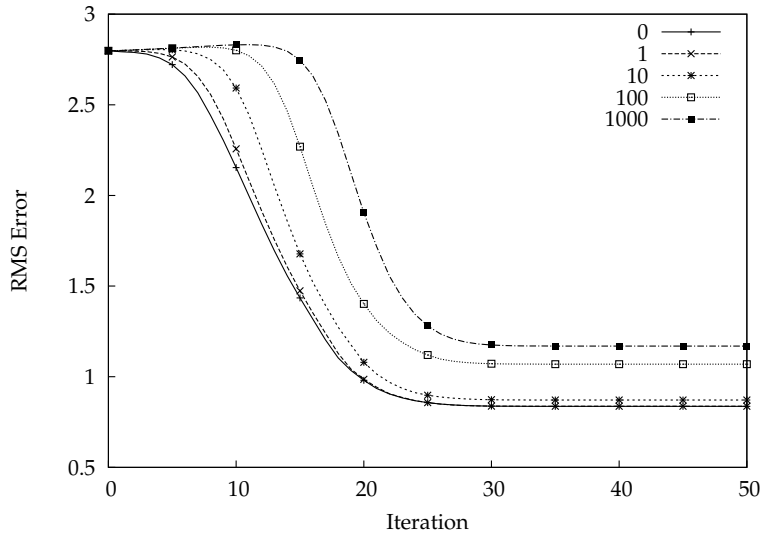


Figure 5.8: The influence of the control parameter k_s

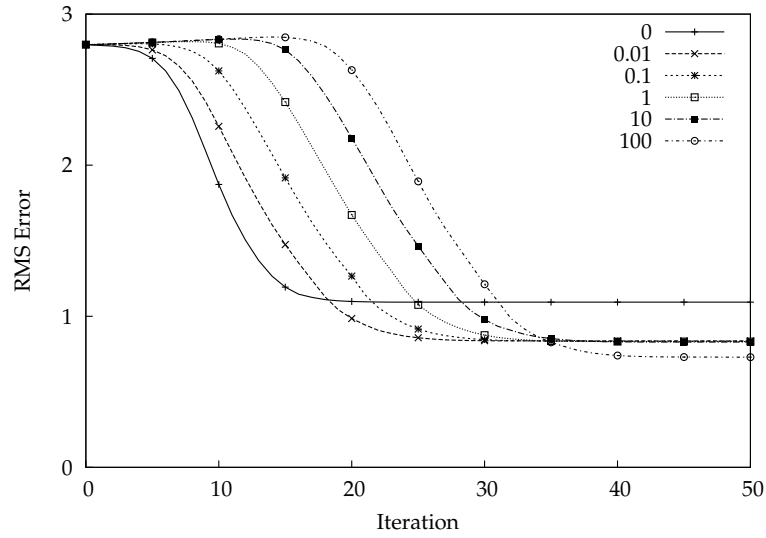
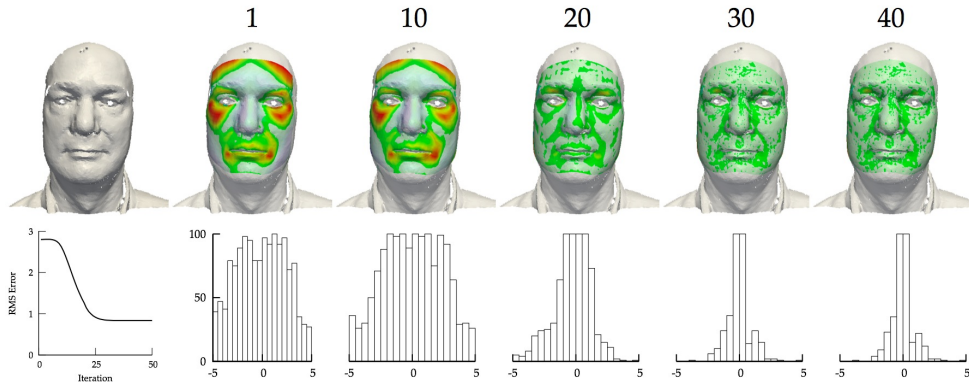
Figure 5.9: The influence of the control parameter k_b 

Figure 5.10: An example deformation process.

5.3 Facial Expression Transferring

Since facial expressions are subjective to everyone, there is no easy way to define a quantitative measurement for the similarity of facial expressions. For this reason, a qualitative evaluation was conducted to evaluate the facial expression transferring method.

An online questionnaire was designed to ask people to rate the similarity of facial expressions in the original faces and the transferred faces. In order to limit the completion time to about 20 mins, the questionnaire includes 15 target models (6 head model + 9 reconstructed model). For each

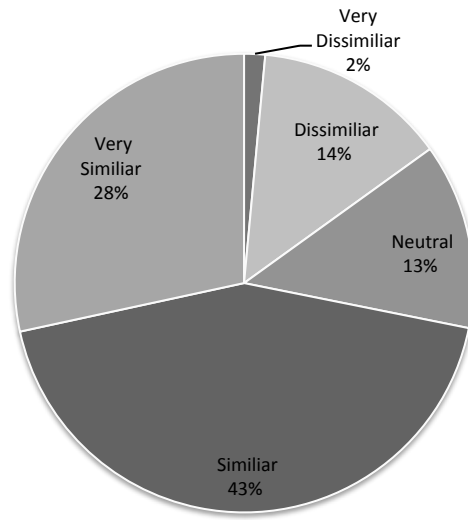


Figure 5.11: Overall rating for facial expression transferring

target model, 6 facial expressions (anger, laughing, pleased, rage, sad and surprise) are transferred from the template model to the target model. A paper version of the questionnaire is shown in Appendix D.

There are 34 volunteers of this qualitative evaluation, who are 2nd, 3rd and 4th year students studying the course of Computer Graphics and Animation at the School of Computing Science and Digital Media. Among them, 32 people completed the online questionnaire.

The overall rating for the similarity of all transferred facial expressions is shown in Figure 5.11. From this figure, we can see that, the number of votes for similar and very similar options is about three quarters of all votes and the number of votes for dissimilar and very dissimilar options is about 16%. It indicates that users considered most of transferred facial expressions similar to the original ones.

Figure 5.12 shows the ratings for each individual facial expression. It is interesting to see that the laughing, pleased and sad expressions have better ratings than that of the anger, rage, and surprise expressions. The laughing, pleased and sad expressions are easier to be recognised and compared because of their universality and relatively large facial motions.

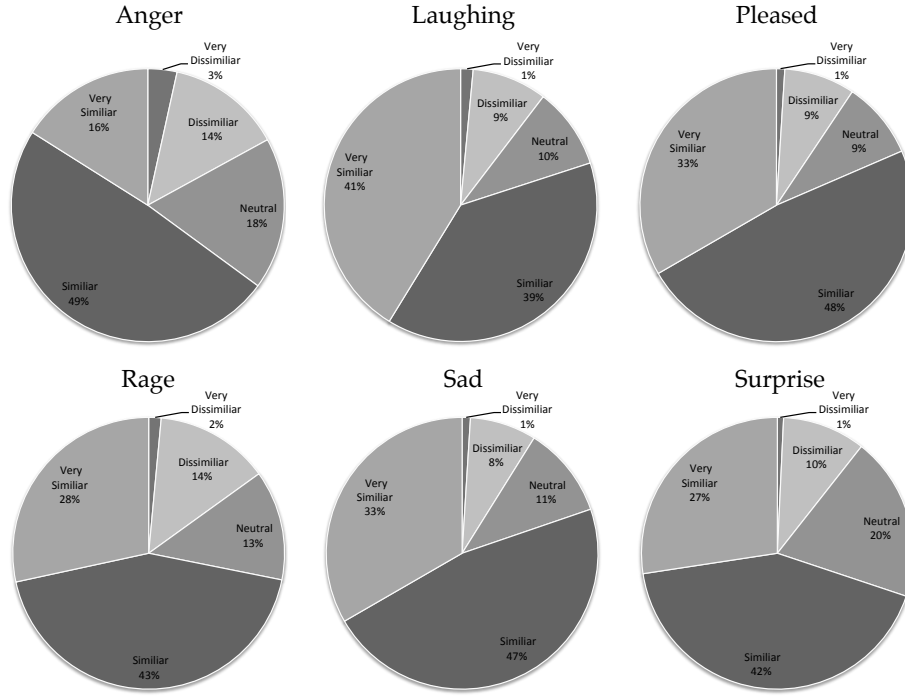


Figure 5.12: Rating for each facial expressions

5.4 Chapter Summary

In this chapter, the quantitative evaluation of the registration method has been tested on synthetic data and public available dataset. The results have shown that the registration method is able to handle certain outliers in range scans and it also comparable to other registration methods. The deformable model and its control parameters have been further discussed. The step shape of the RMS error curve provides a way for optimising the deformable model. Finally, the qualitative evaluation of the facial expression transferring method has been presented. The results have demonstrated the facial expression transferring method can transfer expressions from one model to another.

Conclusions and Future Work

In this chapter, the achievements and contributions of this research are first presented, and then the directions of future work are pointed out.

6.1 Achievements and Contributions

Facial modelling and animation holds tremendous potential to enrich education, human communication, perception and entertainment. However, creating a convincing facial model and animation is a time consuming and difficult process that requires both artistic talent and technical expertise. The work addressed in this thesis eases the process by offering automatic approaches and opportunities for reuse.

This research has investigated the template-based facial modelling, facial expression transferring and facial expression tracking. The following are the most important contributions made over the course of the research:

- **Extension of the ICP algorithm:** In this thesis, an extended of the ICP algorithm has been developed. This registration algorithm is capable of aligning a template with range scans in different scales. The results have demonstrated that the registration method is robust with regard to initial scales and rotations.
- **Deformable model based on the elastic theory of thin-shells:** This thesis has presented a deformable model based on the elastic theory of thin-shells. This deformable model can smoothly deform a mesh

towards its target by adjusting two control parameters. The results have shown that the deformable model can be used to recover the shapes of range scans and to establish dense correspondences across models.

- **Facial model and texture reconstruction from range scans:** Based on the developed registration method and deformable model of this thesis, a fully automatic approach to reconstructing the surfaces and textures of range scans has been proposed. Due to hole filling and smoothing of noisy data, reconstructed facial models are likely to be more realistic than the original range scans. Furthermore, all reconstructed facial models shared the same topological structures as the template facial model. Thus, one-to-one dense correspondences are naturally established across these models, enabling us to do further analysis of the models such as facial expression analysis.
- **Facial expression transferring:** This thesis has also presented a new automatic approach to transferring facial expressions from one facial model to another. This approach is formulated within the framework of discrete different geometry. It encodes facial expressions in displacement deformation gradients instead of displacement vectors. This representation of facial expression is scale-invariant so that facial expressions can be easily cloned in different sizes. This approach reuses facial expressions created in a facial model to quickly generate similar facial expressions on different individuals and thus reduces tremendous repeated work.
- **Facial expression tracking:** Finally, an approach to tracking 3D facial expressions exhibited in a sequence of range scans has been proposed. The results have shown that this approach is able to track the facial expressions which do not exhibit fast and abrupt motions.

6.2 Future Work

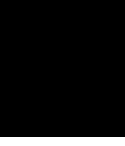
Further research work would be in the following directions:

- **Facial feature detection and matching:** Even though the registration approach developed in this research was robust to initial scales and

rotations, it still required roughly known initial transformation states. There are many salient features in faces, such as nose tip, mouth and eye corners, which could be detected by many feature descriptors (Lowe, 2004; Gal and Cohen-Or, 2006; Zou et al., 2008). Thus, the registration approach could be further extended to be a global registration, which does not require initial transformation states, by detecting and matching these facial features for the estimation of initial transformations. In addition, additional energy terms of feature matching constraints can be integrated into the deformable model.

- **Improvement of facial expression tracking:** The present implementation of the facial expression tracking method only considered the geometric information of range scans. However, the texture images of range scans also contain rich information about facial motions. The well-developed 2D tracking methods, such as optical flow (DeCarlo and Metaxas, 2000), could provide constraints from 2D texture images to the tracking method. The combination of 2D and 3D tracking would improve the tracking accuracy of the tracking method. Furthermore, more constraints could be added to the tracking method by integrating facial feature detection and matching techniques.
- **4D facial expression transferring:** It is possible to combine facial expression transferring with 4D facial expression tracking so that a sequence of facial expressions from 4D range scans could be transferred to another facial model. Firstly, we can establish one-to-one dense correspondences across all the frames of a range scan sequence by using the facial expression tracking method, and then we encode all the facial expressions in the sequences with respect to a reference frame. Next, we establish correspondences between the reference frame and a new facial model by using the deformable model. Finally, we can transfer the facial expressions from the sequence to the new facial model by using the facial expression transferring method.

This page is intentionally left blank.



Triangle Meshes

Due to the simplicity, triangle meshes are extensively used to define shapes of objects in 3D computer graphics. There are many formal descriptions of triangle meshes in the literature (Hoppe et al., 1993; Botsch et al., 2007; Hormann et al., 2008). In order to use consistent notations about triangle meshes in this thesis, this appendix presents a formal mathematical description of triangle meshes based on the paper by Hoppe et al. (1993).

Intuitively, a *triangle mesh* is a piecewise linear surface, consisting of triangular faces pasted together along their edges. It has a topological component and a geometric component. Formally, a triangle mesh \mathcal{M} is a pair of $(\mathcal{K}, \mathcal{P})$, where: \mathcal{K} is a simplicial complex representing the connectivity of the vertices, edges, and faces, thus determining the topological type of the mesh; \mathcal{P} is a set of vertex positions defining the shape of the mesh.

A.1 Simplicial Complexes

A *simplicial complex* \mathcal{K} consists of a set of vertices $\{v_i, i = 1, \dots, n\}$ and a set of non-empty subsets of the vertices called simplexes such that (Spanier, 1994)

- (a) Any set consisting of exactly one vertex is a simplex.
- (b) Every non-empty subset of a simplex is a simplex.

A simple s containing exactly $q + 1$ vertices is called q -simplex. Thus, the 0, 1, 2-simplexes of \mathcal{K} correspond to the vertices, edges, and faces (or trian-

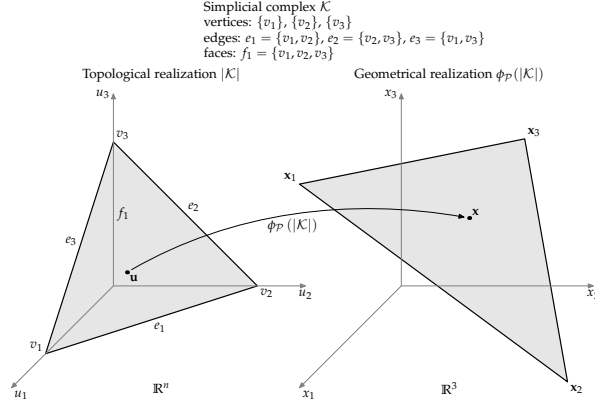


Figure A.1: An example of the mesh representation for a mesh with a single face. The left is the topological realisation and the right is the geometrical realisation

gles) of \mathcal{K} , respectively.

A.2 Topological and Geometrical Realisations

Given a simplicial complex \mathcal{K} , we can identify the vertices $\{v_i, i = 1, \dots, n\}$ with the standard basis vectors $\{\mathbf{u}_i, i = 1, \dots, n\}$ of \mathbb{R}^n , which forms its *topological realisation* $|\mathcal{K}|$ in \mathbb{R}^n . For each simplex $s \in \mathcal{K}$ let $|s|$ denote the convex hull of its vertices in \mathbb{R}^n , and let $|\mathcal{K}| = \cup_{s \in \mathcal{K}} |s|$. Let $\phi: \mathbb{R}^n \rightarrow \mathbb{R}^3$ be the linear map that sends the i -th standard vector $\mathbf{u}_i \in \mathbb{R}^n$ to $\mathbf{x}_i \in \mathbb{R}^3$ (See Figure A.1).

The *geometric realisation* of \mathcal{M} is the image $\phi_{\mathcal{P}}(|\mathcal{K}|)$, where we write the map as $\phi_{\mathcal{P}}$ to emphasise that it is fully specialised by the set of vertex positions $\mathcal{P} = \{\mathbf{x}_i, i = 1, \dots, n\}$. The map $\phi_{\mathcal{P}}$ is called an *embedding* if it is one-to-one, that is if $\phi_{\mathcal{P}}$ is not self-intersecting.

If $\phi_{\mathcal{P}}$ is an embedding, any point $\mathbf{x} \in \phi_{\mathcal{P}}(|\mathcal{K}|)$ can be parameterised by finding its unique pre-image on $|\mathcal{K}|$. The vector $\mathbf{u} \in |\mathcal{K}|$ with $\mathbf{x} \in \phi_{\mathcal{P}}(\mathbf{u})$ is called the *barycentric coordinate vector* of \mathbf{x} (with respect to the simplicial complex \mathcal{K}). Note that barycentric coordinate vectors are convex combination of standard basis vectors $\mathbf{u}_i \in \mathbb{R}^n$ corresponding to the vertices of a face of \mathcal{K} , and only one if it is a vertex.

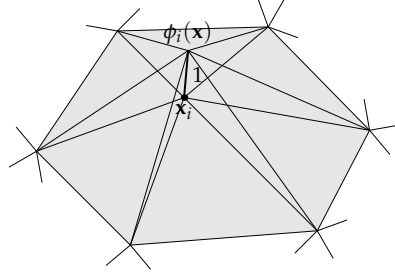


Figure A.2: An illustration of the piecewise linear basis function $\phi_i(\mathbf{x})$ around the 1-ring neighbourhood of a vertex \mathbf{x}_i .

A.3 Piecewise Linear Basis Functions

In practice, however, it is convenient to let the topological realisation of \mathcal{M} be identical to its geometric realisation, *i.e.*, we identify the vertices $\{v_i, i = 1, \dots, n\}$ with the set of vertex positions \mathcal{P} in \mathbb{R}^3 . Then the barycentric coordinates can be defined through a set of *piecewise linear basis functions* $\{\phi_i(\mathbf{x}), i = 1, \dots, n\}$ on the mesh (Figure A.2), such that

$$\sum_{i=1}^n \phi_i(\mathbf{x}) = 1, \quad (\text{A.1})$$

and

$$\phi_i(\mathbf{x}_j) = \delta_{ij}, \quad (\text{A.2})$$

where δ_{ij} is the Kronecker delta given by

$$\delta_{ij} = \begin{cases} 1, & \text{if } i = j; \\ 0, & \text{if } i \neq j. \end{cases} \quad (\text{A.3})$$

If a set of discrete values $\{f_i, i = 1, \dots, n\}$ is defined at the vertices of \mathcal{M} , we can now write a piecewise linear approximation of the values at any point \mathbf{x} on \mathcal{M} by

$$f(\mathbf{x}) = \sum_{i=1}^n f_i \phi_i(\mathbf{x}). \quad (\text{A.4})$$

Especially, the mesh itself can be expressed as a linear combination of its vertex positions, *i.e.*,

$$\mathbf{x} = \sum_{i=1}^n \mathbf{x}_i \phi_i(\mathbf{x}). \quad (\text{A.5})$$

Further discussion about linear basis functions and related differential operators on triangle meshes is presented in Appendix B.

A.4 Two-manifold and Neighbourhoods

A triangle mesh is *2-manifold*, if it does not contain non-manifold edges, non-manifold vertices, or self-intersections. A non-manifold edge has more than two incident triangles and a non-manifold vertex is generated by pinching two surface sheets together at that vertex such that the vertex is incident to two fans of triangles. Non-manifold meshes are problematic for most geometry processing algorithms, since around non-manifold configurations there exists no well-defined local geodesic neighbourhood. Thus, only 2-manifold meshes are considered in this thesis.

For a 2-manifold triangle mesh, the *1-ring vertex neighbourhood* of a vertex v_i is all the vertices connected with the vertex, *i.e.*,

$$\mathcal{N}_1^v(v_i) = \{v_j \mid \{v_i, v_j\} \in \mathcal{K}\}. \quad (\text{A.6})$$

The *1-ring triangle neighbourhood* of a vertex v_i is all the triangles connected with the vertex, *i.e.*,

$$\mathcal{N}_1^t(v_i) = \{t_j \mid v_i \in t_j, t_j \text{ is a 2-simplex}\}. \quad (\text{A.7})$$

Discrete Differential Operators

In this appendix, common differential operators, such as the gradient operator $\nabla_{\mathbf{x}}$, the divergence operator div , and the Laplace-Beltrami operator $\Delta_{\mathbf{x}}$, are generalised from smooth 2-manifold surfaces to triangle meshes. Let us consider a triangle mesh $\mathcal{M} = (\mathcal{K}, \mathcal{P})$, where the topological component \mathcal{K} contains n vertices $\{v_i, i = 1, \dots, n\}$ and m triangles $\{t_j, j = 1, \dots, m\}$, and the vertex positions $\mathcal{P} = \{\mathbf{x}_i, i = 1, \dots, n\}$. In the following text, we will first define discrete potential and gradient fields on the triangle mesh \mathcal{M} and then deduce the discrete differential operators applying on these fields.

B.1 Discrete Potential Fields

A *discrete scalar potential field* on the triangle mesh \mathcal{M} is a set of scalars $\{f_i, i = 1, \dots, n\}$ with each scalar f_i one-to-one corresponding to a vertex v_i . Given a set of piecewise linear basis functions $\{\phi_i(\mathbf{x}), i = 1, \dots, n\}$ defined on the triangle mesh \mathcal{M} , we can write a piecewise linear approximation of the scalar field at any point \mathbf{x} on the triangle mesh \mathcal{M} as

$$f(\mathbf{x}) = \sum_{i=1}^n f_i \phi_i(\mathbf{x}). \quad (\text{B.1})$$

Similarly, if each vertex v_i of the triangle mesh \mathcal{M} is associated with a vector \mathbf{f}_i , then the set of the vectors $\{\mathbf{f}_i, i = 1, \dots, n\}$ forms a *discrete vector potential field* on the triangle mesh \mathcal{M} . A linear approximation of the vector

field is

$$\mathbf{f}(\mathbf{x}) = \sum_{i=1}^n \mathbf{f}_i \phi_i(\mathbf{x}). \quad (\text{B.2})$$

For example, the set of the vertex positions \mathcal{P} is a discrete vector potential field on the triangle mesh \mathcal{M} . If the components of a discrete vector potential field are independent of each other, then each component of the field forms a discrete scalar potential field.

B.2 Discrete Gradient Fields

A *discrete gradient field* on the triangle mesh \mathcal{M} is set of vectors $\{\mathbf{g}_j, j = 1, \dots, m\}$ such that each vector \mathbf{g}_j one-to-one corresponds to a triangle t_j and is *coplanar* with the triangle. Given a set of piecewise constant basis functions $\{\psi_j(\mathbf{x}), j = 1, \dots, m\}$ satisfying

$$\psi_j(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{x} \in |t_j|, \\ 0, & \text{if } \mathbf{x} \notin |t_j|, \end{cases} \quad (\text{B.3})$$

where $|t_j|$ denotes the convex hull of a triangle t_j in \mathbb{R}^3 , we can express the discrete gradient field as a function

$$\mathbf{g}(\mathbf{x}) = \sum_{j=1}^m \mathbf{g}_j \psi_j(\mathbf{x}). \quad (\text{B.4})$$

B.3 Discrete Gradient Operator

Given the above definitions of the discrete potential and gradient fields on a triangle mesh, we can now deduce the discrete gradient operator on the triangle mesh \mathcal{M} . For a triangle t_j , let us denote its linear basis functions and scalars by ϕ_{jk} and f_{jk} , $k = 1, 2, 3$, respectively. The first subscript denotes a triangle index and the second a local vertex index of the triangle. Then we can rewrite Equation B.1 as

$$f(\mathbf{x}) = \sum_{j=1}^m f_j(\mathbf{x}) = \sum_{j=1}^m \sum_{k=1}^3 f_{jk} \phi_{jk}(\mathbf{x}), \quad (\text{B.5})$$

where $f_j(\mathbf{x})$ is a linear function defined on the triangle t_j . The gradient of the scalar function $f(\mathbf{x})$ with respect to the spatial coordinates \mathbf{x} of the piecewise

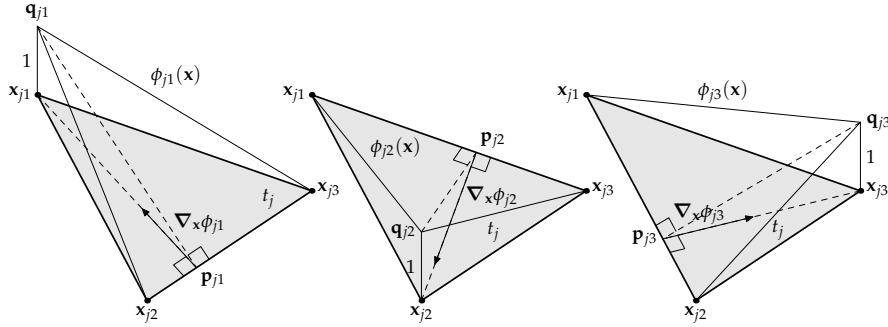


Figure B.1: A geometrical illustration of the linear basis functions $\phi_{jk}(\mathbf{x})$ and their corresponding gradients $\nabla_{\mathbf{x}}\phi_{jk}(\mathbf{x})$ on a triangle t_j .

linear surface of the triangle mesh \mathcal{M} is

$$\nabla_{\mathbf{x}}f(\mathbf{x}) = \sum_{j=1}^m \sum_{k=1}^3 f_{jk} \nabla_{\mathbf{x}}\phi_{jk}(\mathbf{x}), \quad (\text{B.6})$$

where $\nabla_{\mathbf{x}}$ denotes $\partial/\partial\mathbf{x}$, the first-order partial differential operator with respect to the spatial coordinates \mathbf{x} of the piecewise linear surface of the triangle mesh \mathcal{M} .

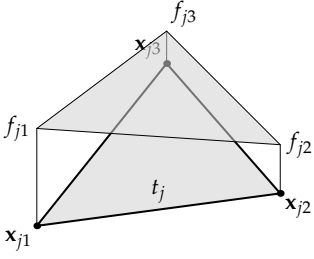
The gradients of the linear basis functions can be directly obtained from geometrical considerations. From Figure B.1, we can see that the gradient of the basis functions $\phi_{j1}(\mathbf{x})$ is a constant vector with the magnitude of $1/\|\mathbf{x}_{j1} - \mathbf{p}_{j1}\|$, which is perpendicular to the vector $\mathbf{x}_{j3} - \mathbf{x}_{j2}$ and points to \mathbf{x}_{j1} , *i.e.*,

$$\begin{aligned} \nabla_{\mathbf{x}}\phi_{j1}(\mathbf{x}) &= \frac{\mathbf{x}_{j1} - \mathbf{p}_{j1}}{\|\mathbf{x}_{j1} - \mathbf{p}_{j1}\|^2} \\ &= \frac{\|\mathbf{x}_{j3} - \mathbf{x}_{j2}\|}{\|\mathbf{x}_{j1} - \mathbf{p}_{j1}\| \|\mathbf{x}_{j3} - \mathbf{x}_{j2}\|} \frac{\mathbf{x}_{j1} - \mathbf{p}_{j1}}{\|\mathbf{x}_{j1} - \mathbf{p}_{j1}\|} \\ &= \frac{\mathbf{n}_j \times (\mathbf{x}_{j3} - \mathbf{x}_{j2})}{2A_j}, \end{aligned} \quad (\text{B.7})$$

where A_j and \mathbf{n}_j are the area and the normal of the triangle t_j , respectively. Similarly, the gradients of the other two basis functions are

$$\nabla_{\mathbf{x}}\phi_{j2}(\mathbf{x}) = \frac{\mathbf{n}_j \times (\mathbf{x}_{j1} - \mathbf{x}_{j3})}{2A_j}, \quad (\text{B.8})$$

$$\nabla_{\mathbf{x}}\phi_{j3}(\mathbf{x}) = \frac{\mathbf{n}_j \times (\mathbf{x}_{j2} - \mathbf{x}_{j1})}{2A_j}. \quad (\text{B.9})$$



$$\mathbf{g}_j = (\nabla_{\mathbf{x}}\phi_{j1}(\mathbf{x}) \nabla_{\mathbf{x}}\phi_{j2}(\mathbf{x}) \nabla_{\mathbf{x}}\phi_{j3}(\mathbf{x})) \begin{pmatrix} f_{j1} \\ f_{j2} \\ f_{j3} \end{pmatrix}$$

$$\begin{pmatrix} \mathbf{g}_1 \\ \vdots \\ \mathbf{g}_j \\ \vdots \\ \mathbf{g}_m \end{pmatrix}_{3m \times 1} = \begin{pmatrix} t_1 & & & & \\ & \ddots & & & \\ & & \nabla_{\mathbf{x}}\phi_{j1}(\mathbf{x}) & \cdots & \nabla_{\mathbf{x}}\phi_{j2}(\mathbf{x}) & \cdots & \nabla_{\mathbf{x}}\phi_{j3}(\mathbf{x}) & \\ & & & & & & & \ddots \\ t_m & & & & & & & \end{pmatrix} \begin{pmatrix} f_1 \\ \vdots \\ f_{j1} \\ \vdots \\ f_{j2} \\ \vdots \\ f_{j3} \\ \vdots \\ f_n \end{pmatrix}_{n \times 1}$$

$\mathcal{G}_{3m \times n}$

Figure B.2: The assembly of the sparse matrix \mathcal{G} , called the discrete gradient operator, for the triangle t_j .

As a result, Equation B.6 can be rewritten in the following matrix form (See Figure B.2 below)

$$\mathbf{g} = \mathcal{G}\mathbf{f}, \quad (\text{B.10})$$

where \mathbf{f} is an n -dimensional column vector representing a discrete scalar potential field, \mathbf{g} is a $3m$ -dimensional column vector representing the gradient of the discrete scalar potential field with respect to \mathbf{x} , and \mathcal{G} is a $3m \times n$ sparse matrix. The sparse matrix \mathcal{G} , called the *discrete gradient operator*, transforms a discrete scalar potential field into a discrete gradient field. Since it only depends on the topological and geometrical structures of a mesh, the sparse matrix \mathcal{G} can be pre-computed and stored for any future reuse.

B.4 Discrete Divergence Operator

Now consider a discrete gradient field $\{\mathbf{g}_j, j = 1, \dots, m\}$ defined on the triangle mesh \mathcal{M} . The goal is to compute the divergence of the discrete

gradient field at each vertex of the triangle mesh \mathcal{M} .

Given a piecewise constant function $\mathbf{g}(\mathbf{x})$ of the discrete gradient field as expressed by Equation B.4, we cannot apply the divergence operator on it directly, otherwise we will get zero within triangles and the infinite along edges. Instead, we employ a spatial average approach, proposed in the paper by Meyer et al. (2002), to approximate the divergence at each vertex.

The spatial average equation for the divergence of the gradient field is

$$\nabla_{\mathbf{x}} \cdot \mathbf{g}(\mathbf{x}) \approx \lim_{A \rightarrow 0} \frac{1}{A} \int_{\mathcal{A}} \nabla_{\mathbf{x}} \cdot \mathbf{g}(\mathbf{x}) dA, \quad (\text{B.11})$$

where \mathcal{A} is a properly selected region around \mathbf{x}_i and A is the area of the region \mathcal{A} .

Meyer et al. (2002) showed that using a Voronoi region \mathcal{A}_V provides provably tight error bounds for non-obtuse triangle meshes. For arbitrary meshes containing obtuse triangles, a mixed-type region \mathcal{A}_M , which contains circumcenters of non-obtuse triangles and midpoints of the longest edges of obtuse triangles, is used (see the inner grey region in Figure B.3). Therefore, we have

$$\nabla_{\mathbf{x}} \cdot \mathbf{g}(\mathbf{x}) \approx \frac{1}{A_M} \int_{\mathcal{A}_M} \nabla_{\mathbf{x}} \cdot \mathbf{g}(\mathbf{x}) dA, \quad (\text{B.12})$$

where A_M is the area of the mixed local support region \mathcal{A}_M around \mathbf{x}_i .

Using Gauss' theorem, we can turn the integral of a divergence over a region into a line integral over the boundary of the region, *i.e.*,

$$\frac{1}{A_M} \int_{\mathcal{A}_M} \nabla_{\mathbf{x}} \cdot \mathbf{g}(\mathbf{x}) dA = \frac{1}{A_M} \int_{\partial \mathcal{A}_M} \boldsymbol{\nu} \cdot \mathbf{g}(\mathbf{x}) ds, \quad (\text{B.13})$$

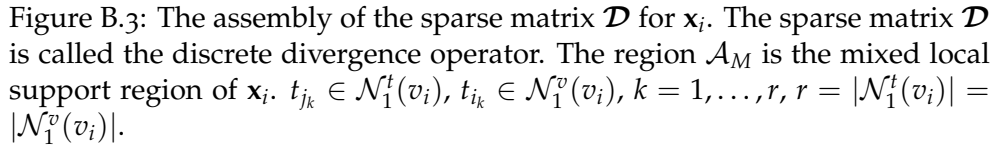
where $\boldsymbol{\nu}$ is the outer normal of the boundary $\partial \mathcal{A}_M$.

Consider a 1-ring neighbouring triangle $t_{j_k} \in \mathcal{N}_1^t(v_i)$ of v_i (See Figure B.3). Without loss of generality, denote the other two vertices of t_{j_k} by $v_{i_k}, v_{i_{k+1}} \in \mathcal{N}_1^v(v_i)$, the gradient on t_{j_k} by \mathbf{g}_{j_k} . Then the line integral of the right side of Equation B.13 inside the triangle t_{j_k} becomes

$$\int_{\partial \mathcal{A}_M \cap |t_{j_k}|} \boldsymbol{\nu} \cdot \mathbf{g}(\mathbf{x}) ds = \int_{\overline{\mathbf{p}\mathbf{r}}} \boldsymbol{\nu}_1 \cdot \mathbf{g}_{j_k} ds + \int_{\overline{\mathbf{r}\mathbf{q}}} \boldsymbol{\nu}_2 \cdot \mathbf{g}_{j_k} ds. \quad (\text{B.14})$$

Since the vector \mathbf{g}_{j_k} is constant inside t_{j_k} , the circular integral of \mathbf{g}_{j_k} along the closed path $\mathbf{p} \rightarrow \mathbf{r} \rightarrow \mathbf{q} \rightarrow \mathbf{p}$ is zero. Therefore, the above equation becomes

$$\int_{\partial \mathcal{A}_M \cap |t_{j_k}|} \boldsymbol{\nu} \cdot \mathbf{g}(\mathbf{x}) ds = \int_{\overline{\mathbf{p}\mathbf{q}}} \boldsymbol{\nu}_3 \cdot \mathbf{g}_{j_k} ds = \mathbf{n}_{j_k} \times (\mathbf{p} - \mathbf{q}) \cdot \mathbf{g}_{j_k}, \quad (\text{B.15})$$



Note that \mathbf{p} and \mathbf{q} are the midpoints of the two edges. Thus, we have

Substituting Equations B.16 and B.7 into Equation B.15, we obtain

where A_{j_k} is the area of the triangle t_{j_k} , and $\phi_i^{j_k}(\mathbf{x})$ denotes the basic function $\phi_i(\mathbf{x})$ within the triangle t_{j_k} .

Finally, the divergence of the discrete vector field $\mathbf{g}(\mathbf{x})$ at the vertex \mathbf{x}_i

can be written as

$$\begin{aligned}
\nabla_{\mathbf{x}} \cdot \mathbf{g}(\mathbf{x}) &\approx \frac{1}{A_M} \int_{\partial \mathcal{A}_M} \boldsymbol{\nu} \cdot \mathbf{g}(\mathbf{x}) ds \\
&= \frac{1}{A_M} \sum_{t_{jk} \in \mathcal{N}_1^t(v_i)} \int_{\partial \mathcal{A}_M \cap |t_{jk}|} \boldsymbol{\nu} \cdot \mathbf{g}(\mathbf{x}) ds \\
&= \sum_{t_{jk} \in \mathcal{N}_1^t(v_i)} -\frac{A_{j_k}}{A_M} \nabla_{\mathbf{x}} \phi_i^{j_k}(\mathbf{x}) \cdot \mathbf{g}_{j_k}, \tag{B.18}
\end{aligned}$$

or in the matrix form (See Figure B.3),

$$\mathbf{f} = \mathcal{D} \mathbf{g}, \tag{B.19}$$

where \mathbf{g} is a $3m$ -dimensional column vector representing a discrete gradient field, \mathbf{f} is an n -dimensional column vector representing the divergence of the discrete gradient field, and \mathcal{D} is an $n \times 3m$ sparse matrix. The sparse matrix \mathcal{D} , called the *discrete divergence operator*, transforms a discrete gradient field into a discrete scalar potential field. This sparse matrix can also be pre-computed.

B.5 Discrete Laplace-Beltrami Operator

The Laplace-Beltrami operator on smooth surfaces is defined as

$$\Delta_{\mathbf{x}} = \nabla_{\mathbf{x}} \cdot \nabla_{\mathbf{x}}, \tag{B.20}$$

which is the divergence of the gradient. Analogously, we can define the discrete Laplace-Beltrami operator \mathcal{L} as follows,

$$\mathcal{L} = \mathcal{D} \mathcal{G}, \tag{B.21}$$

where \mathcal{D} and \mathcal{G} are the discrete divergence and gradient operators that are deduced in the above sections. We will show that the above equation yields the same results as that obtained by Meyer et al. (2002).

From Figures B.2 and B.3, the discrete Laplace-Beltrami operator within the local support region \mathcal{A}_M around \mathbf{x}_i is

$$\Delta_{\mathbf{x}}|_{\mathbf{x}_i} = \sum_{t_{jk} \in \mathcal{N}_1^t(v_i)} -\frac{A_{j_k}}{A_M} \left(\nabla_{\mathbf{x}} \phi_i^{j_k} \right)^T \begin{pmatrix} \nabla_{\mathbf{x}} \phi_i^{j_k} & \nabla_{\mathbf{x}} \phi_{i_k}^{j_k} & \nabla_{\mathbf{x}} \phi_{i_{k+1}}^{j_k} \end{pmatrix}. \tag{B.22}$$

Since the sum of the linear basis functions is 1, thus we have

$$\nabla_{\mathbf{x}}\phi_i^{j_k} + \nabla_{\mathbf{x}}\phi_{i_k}^{j_k} + \nabla_{\mathbf{x}}\phi_{i_{k+1}}^{j_k} = 0. \quad (\text{B.23})$$

Using the Binet-Cauchy identity

$$(\mathbf{a} \times \mathbf{b}) \cdot (\mathbf{c} \times \mathbf{d}) = (\mathbf{a} \cdot \mathbf{c})(\mathbf{b} \cdot \mathbf{d}) - (\mathbf{a} \cdot \mathbf{d})(\mathbf{b} \cdot \mathbf{c}), \quad (\text{B.24})$$

we obtain

$$\begin{aligned} \left(\nabla_{\mathbf{x}}\phi_i^{j_k}\right)^T \nabla_{\mathbf{x}}\phi_{i_k}^{j_k} &= \frac{\mathbf{n}_{j_k} \times (\mathbf{x}_{i_{k+1}} - \mathbf{x}_{i_k})}{2A_{j_k}} \cdot \frac{\mathbf{n}_{j_k} \times (\mathbf{x}_i - \mathbf{x}_{i_k})}{2A_{j_k}} \\ &= -\frac{1}{2A_{j_k}} \frac{(\mathbf{x}_{i_k} - \mathbf{x}_{i_{k+1}}) \cdot (\mathbf{x}_i - \mathbf{x}_{i_{k+1}})}{2A_{j_k}} \\ &= -\frac{1}{2A_{j_k}} \frac{\|\mathbf{x}_{i_k} - \mathbf{x}_{i_{k+1}}\| \|\mathbf{x}_i - \mathbf{x}_{i_{k+1}}\| \cos \beta_k}{\|\mathbf{x}_{i_k} - \mathbf{x}_{i_{k+1}}\| \|\mathbf{x}_i - \mathbf{x}_{i_{k+1}}\| \sin \beta_k} \\ &= -\frac{\cot \beta_k}{2A_{j_k}}. \end{aligned} \quad (\text{B.25})$$

Similarly,

$$\left(\nabla_{\mathbf{x}}\phi_i^{j_k}\right)^T \nabla_{\mathbf{x}}\phi_{i_{k+1}}^{j_k} = -\frac{\cot \alpha_k}{2A_{j_k}}. \quad (\text{B.26})$$

Substituting Equations B.23, B.25 and B.26 into Equation (B.17), we get

$$\Delta_{\mathbf{x}}f|_{\mathbf{x}_i} = \sum_{t_{j_k} \in \mathcal{N}_1^t(v_i)} \frac{1}{2A_M} \begin{pmatrix} -(\cot \alpha_k + \cot \beta_k) & \cot \beta_k & \cot \alpha_k \end{pmatrix} \quad (\text{B.27})$$

Thus, for a given discrete scalar potential field $\{f_i, i = 1, \dots, n\}$ and its piecewise linear approximation $f(\mathbf{x})$, the Laplace-Beltrami of the field at the vertex \mathbf{x}_i is

$$\begin{aligned} \Delta_{\mathbf{x}}f(\mathbf{x})|_{\mathbf{x}_i} &= \sum_{t_{j_k} \in \mathcal{N}_1^t(v_i)} \frac{1}{2A_M} \begin{pmatrix} -(\cot \alpha_k + \cot \beta_k) & \cot \beta_k & \cot \alpha_k \end{pmatrix} \begin{pmatrix} f_i \\ f_{i_k} \\ f_{i_{k+1}} \end{pmatrix} \\ &= \frac{1}{2A_M} \sum_{t_{j_k} \in \mathcal{N}_1^t(v_i)} \cot \alpha_k (f_{i_{k+1}} - f_i) + \cot \beta_k (f_{i_k} - f_i) \\ &= \frac{1}{2A_M} \sum_{v_{i_k} \in \mathcal{N}_1^g(v_i)} (\cot \alpha_{k-1} + \cot \beta_k) (f_{i_k} - f_i), \end{aligned} \quad (\text{B.28})$$

which is the same as that obtained by Meyer et al. (2002).

B.6 Laplacian Smoothing

Laplacian smoothing is a simple and very effective technique based on linear diffusion of vertex positions: the vertex positions are subject to diffusion, which equilibrates small differences (noise). Mathematically, the linear diffusion of vertex positions is

$$\frac{\partial \mathbf{x}}{\partial t} = \mu \Delta \mathbf{x}, \quad (\text{B.29})$$

where \mathbf{x} is the vertex position, μ the diffusion constant, and Δ the Laplacian operator. For triangle meshes, the above equation can be discretised based on finite differences and the discretisation of the Laplacian operator, and approximated by the iterative equation

$$\mathbf{x}_i^{(k+1)} = \mathbf{x}_i^{(k)} + \mu \sum_{j \in \mathcal{N}_1(i)} w_{ij} (\mathbf{x}_j^{(k)} - \mathbf{x}_i^{(k)}), \quad (\text{B.30})$$

where $\mathbf{x}_i^{(k)}$ denotes the k -th iteration of the i -th vertex position with $\mathbf{x}_i^{(0)} = \mathbf{x}$, and $\mathcal{N}_1(\cdot)$ is the 1-ring neighbourhood around a vertex. The Laplacian weight w_{ij} for the edge connecting vertices \mathbf{x}_i and \mathbf{x}_j in the equation is

$$w_{ij} = \frac{1}{2} (\cot \alpha + \cot \beta), \quad (\text{B.31})$$

where α and β are the two opposite angles of the edge (See Section B.5 for more details about the Laplacian operator on triangle meshes).

This page is intentionally left blank.

FRGC Test Results

This appendix presents the test results of the evaluation on the FRGC v2.0 dataset, which is discussed in Sections 5.1 and 5.2.

This page is intentionally left blank.

Table C.1: Results of 200 tests on the FRGC dataset

| Range Scan | | | Triangulation | | | Smoothing | | Decimation | | | | Cleanup | | | Registration | | | Deformation | |
|------------|--------|------------|---------------|--------|--------|-----------|------------------|------------|-------|-------|------------------|---------|-------|-------|--------------|------------------|---|-------------|------------------|
| ID | N_p | r_{\max} | t | N_v | N_t | t | e_{rms} | t | N_v | N_t | e_{rms} | t | N_v | N_t | t | e_{rms} | R | t | e_{rms} |
| 04202d452 | 88641 | 213.47 | 3.30 | 88641 | 173488 | 8.02 | 0.158 | 26.20 | 9257 | 16934 | 0.052 | 0.06 | 5658 | 10428 | 1.53 | 3.432 | Y | 2.79 | 1.277 |
| 04314d53 | 96791 | 217.56 | 2.99 | 96791 | 188994 | 9.45 | 0.153 | 27.95 | 10085 | 18446 | 0.059 | 0.07 | 8662 | 15718 | 2.45 | 3.196 | Y | 3.04 | 0.427 |
| 04202d450 | 70370 | 159.93 | 3.24 | 70370 | 138106 | 6.32 | 0.152 | 19.69 | 7277 | 13556 | 0.048 | 0.05 | 5912 | 11108 | 1.66 | 3.837 | Y | 2.80 | 1.505 |
| 04202d456 | 72906 | 174.20 | 3.30 | 72906 | 142986 | 6.78 | 0.148 | 22.21 | 7588 | 13998 | 0.046 | 0.05 | 6716 | 12447 | 1.58 | 3.300 | Y | 2.84 | 1.368 |
| 04202d454 | 77114 | 193.39 | 3.20 | 77114 | 151422 | 6.95 | 0.159 | 22.83 | 7982 | 14848 | 0.052 | 0.05 | 7084 | 12970 | 1.91 | 3.529 | Y | 2.89 | 1.344 |
| 04334d308 | 78175 | 161.70 | 2.91 | 78175 | 152482 | 7.84 | 0.149 | 22.17 | 8146 | 14909 | 0.055 | 0.06 | 6176 | 11544 | 1.70 | 3.030 | Y | 3.12 | 1.076 |
| 04334d302 | 81658 | 162.76 | 2.84 | 81658 | 160130 | 8.08 | 0.151 | 21.85 | 8444 | 15713 | 0.047 | 0.06 | 6306 | 11878 | 1.73 | 3.029 | Y | 3.02 | 0.926 |
| 04334d304 | 105723 | 224.65 | 3.04 | 105723 | 206472 | 10.63 | 0.148 | 30.08 | 11017 | 20198 | 0.051 | 0.07 | 6586 | 12399 | 1.71 | 2.864 | Y | 2.97 | 0.737 |
| 04334d306 | 103338 | 228.83 | 2.94 | 103338 | 202698 | 10.19 | 0.150 | 29.89 | 10691 | 19902 | 0.046 | 0.07 | 6365 | 11848 | 1.65 | 2.736 | N | - | - |
| 04261d299 | 120807 | 190.30 | 2.94 | 120807 | 237516 | 11.55 | 0.135 | 36.30 | 12502 | 23303 | 0.044 | 0.09 | 9151 | 17286 | 2.44 | 3.139 | Y | 2.89 | 0.994 |
| 04297d263 | 130682 | 230.69 | 2.93 | 130682 | 256172 | 12.66 | 0.142 | 39.98 | 13567 | 25089 | 0.048 | 0.10 | 12228 | 22392 | 3.94 | 4.291 | N | - | - |
| 04321d118 | 88910 | 192.71 | 2.86 | 88910 | 174622 | 9.19 | 0.163 | 24.27 | 9202 | 17125 | 0.059 | 0.06 | 8196 | 14860 | 2.20 | 2.092 | Y | 2.97 | 0.587 |
| 04322d136 | 131108 | 221.50 | 2.87 | 131108 | 258326 | 12.91 | 0.143 | 37.78 | 13512 | 25418 | 0.043 | 0.10 | 12549 | 23175 | 3.27 | 2.657 | Y | 3.26 | 0.661 |
| 02463d560 | 101257 | 185.29 | 2.54 | 101257 | 199868 | 8.71 | 0.164 | 32.28 | 10399 | 19698 | 0.052 | 0.07 | 9027 | 17043 | 1.76 | 2.789 | Y | 2.86 | 0.603 |
| 04322d134 | 123402 | 224.54 | 2.81 | 123402 | 242668 | 11.93 | 0.146 | 33.70 | 12762 | 23824 | 0.048 | 0.09 | 11269 | 20760 | 2.87 | 2.566 | Y | 3.18 | 0.561 |
| 02463d562 | 109120 | 190.76 | 2.47 | 109120 | 214936 | 9.55 | 0.153 | 35.87 | 11228 | 21146 | 0.048 | 0.08 | 9425 | 17629 | 1.76 | 2.714 | Y | 2.86 | 0.767 |
| 04322d132 | 115975 | 193.47 | 2.76 | 115975 | 228842 | 11.59 | 0.139 | 31.96 | 11905 | 22561 | 0.042 | 0.10 | 11247 | 20987 | 2.39 | 2.447 | Y | 3.16 | 0.496 |
| 04298d73 | 67858 | 149.93 | 3.08 | 67858 | 133266 | 7.02 | 0.157 | 20.25 | 7009 | 13085 | 0.052 | 0.05 | 6212 | 11402 | 1.64 | 2.711 | Y | 2.86 | 0.758 |
| 04322d130 | 151653 | 252.26 | 2.93 | 151653 | 299144 | 15.58 | 0.145 | 42.84 | 15553 | 29500 | 0.041 | 0.13 | 14690 | 27635 | 4.90 | 6.261 | N | - | - |
| 04287d51 | 91166 | 226.88 | 2.78 | 91166 | 175734 | 8.41 | 0.145 | 26.88 | 9713 | 16906 | 0.055 | 0.06 | 6199 | 11462 | 1.95 | 3.861 | Y | 2.88 | 1.397 |
| 04203d440 | 120355 | 231.79 | 3.38 | 120355 | 236850 | 11.31 | 0.148 | 35.37 | 12431 | 23268 | 0.043 | 0.09 | 11744 | 21839 | 3.77 | 4.815 | N | - | - |
| 04217d411 | 107538 | 232.52 | 3.09 | 107538 | 208706 | 11.29 | 0.155 | 31.49 | 11357 | 20253 | 0.054 | 0.08 | 5308 | 9527 | 2.42 | 2.969 | N | - | - |
| 04203d442 | 107415 | 195.22 | 3.15 | 107415 | 210896 | 10.10 | 0.137 | 31.95 | 11139 | 20666 | 0.045 | 0.08 | 8794 | 16433 | 1.77 | 2.318 | Y | 2.83 | 0.724 |
| 04217d413 | 87876 | 159.69 | 3.07 | 87876 | 170348 | 9.48 | 0.122 | 25.59 | 9282 | 16545 | 0.049 | 0.06 | 6764 | 12465 | 2.37 | 4.438 | Y | 2.86 | 1.560 |
| 04203d444 | 116177 | 214.90 | 3.07 | 116177 | 228782 | 10.70 | 0.136 | 33.83 | 11950 | 22540 | 0.043 | 0.09 | 11125 | 20715 | 2.81 | 2.741 | Y | 3.03 | 1.108 |
| 04203d446 | 106664 | 216.84 | 2.96 | 106664 | 209520 | 9.96 | 0.145 | 32.12 | 11053 | 20542 | 0.044 | 0.08 | 10407 | 18981 | 2.58 | 2.739 | Y | 2.98 | 1.255 |
| 04203d448 | 99970 | 206.92 | 3.10 | 99970 | 195970 | 9.18 | 0.149 | 28.49 | 10392 | 19174 | 0.053 | 0.07 | 8039 | 14892 | 1.67 | 2.270 | Y | 2.83 | 0.899 |
| 04327d296 | 134741 | 214.90 | 2.86 | 134741 | 262370 | 14.24 | 0.168 | 31.41 | 14176 | 25450 | 0.085 | 0.10 | 10966 | 19246 | 3.41 | 4.312 | Y | 3.27 | 2.534 |
| 04387d322 | 90925 | 178.23 | 2.33 | 90925 | 178442 | 6.70 | 0.150 | 20.46 | 9439 | 17486 | 0.047 | 0.07 | 7212 | 13504 | 1.68 | 3.282 | Y | 3.00 | 1.265 |
| 04299d193 | 97963 | 216.01 | 2.93 | 97963 | 190810 | 9.93 | 0.139 | 30.24 | 10269 | 18611 | 0.058 | 0.08 | 9268 | 17064 | 4.34 | 5.844 | N | - | - |
| 04336d297 | 101868 | 192.85 | 2.36 | 101868 | 197968 | 7.39 | 0.154 | 22.71 | 10691 | 19261 | 0.064 | 0.07 | 5802 | 10818 | 1.54 | 1.983 | Y | 2.97 | 0.468 |
| 04288d258 | 94558 | 189.41 | 2.79 | 94558 | 185394 | 9.44 | 0.146 | 28.41 | 9798 | 18193 | 0.052 | 0.07 | 9005 | 16729 | 2.42 | 2.980 | Y | 3.01 | 0.443 |

Table C.1: Results of 200 tests on the FRGC dataset

| Range Scan | | | Triangulation | | | Smoothing | | Decimation | | | | Cleanup | | | Registration | | | Deformation | |
|------------|--------|------------|---------------|--------|--------|-----------|------------------|------------|-------|-------|------------------|---------|-------|-------|--------------|------------------|---|-------------|------------------|
| ID | N_p | r_{\max} | t | N_v | N_t | t | e_{rms} | t | N_v | N_t | e_{rms} | t | N_v | N_t | t | e_{rms} | R | t | e_{rms} |
| 04237d153 | 79293 | 186.08 | 2.72 | 79293 | 154840 | 7.25 | 0.170 | 22.85 | 8276 | 15097 | 0.064 | 0.06 | 5807 | 10707 | 1.80 | 2.512 | Y | 2.74 | 0.725 |
| 04237d151 | 131002 | 251.64 | 2.80 | 131002 | 257816 | 12.36 | 0.168 | 37.93 | 13530 | 25328 | 0.063 | 0.10 | 12525 | 22878 | 4.79 | 5.129 | N | - | - |
| 04237d157 | 81377 | 171.79 | 2.73 | 81377 | 159780 | 7.92 | 0.153 | 28.80 | 8431 | 15658 | 0.053 | 0.06 | 7882 | 14439 | 2.26 | 2.715 | Y | 2.87 | 0.756 |
| 04299d191 | 80601 | 169.13 | 2.98 | 80601 | 156208 | 7.84 | 0.137 | 23.39 | 8466 | 15196 | 0.065 | 0.06 | 7278 | 13204 | 2.31 | 2.634 | Y | 2.92 | 1.062 |
| 04237d155 | 74267 | 167.21 | 2.72 | 74267 | 145784 | 6.78 | 0.166 | 24.52 | 7682 | 14307 | 0.061 | 0.06 | 6756 | 12325 | 1.93 | 2.506 | Y | 2.79 | 0.478 |
| 04428d241 | 89981 | 209.45 | 2.32 | 89981 | 176228 | 6.59 | 0.169 | 21.60 | 9319 | 17257 | 0.059 | 0.07 | 8959 | 16195 | 2.69 | 2.554 | N | - | - |
| 04324d278 | 117998 | 222.03 | 2.65 | 117998 | 230430 | 11.47 | 0.147 | 33.08 | 12348 | 22446 | 0.058 | 0.09 | 7526 | 13670 | 2.10 | 2.666 | Y | 3.00 | 0.791 |
| 04297d269 | 100332 | 205.10 | 2.88 | 100332 | 195892 | 10.62 | 0.144 | 29.34 | 10501 | 19083 | 0.059 | 0.08 | 9903 | 17896 | 2.51 | 2.603 | Y | 2.99 | 0.542 |
| 04225d303 | 105961 | 255.60 | 2.81 | 105961 | 208260 | 11.22 | 0.165 | 31.94 | 10948 | 20458 | 0.055 | 0.08 | 10248 | 18635 | 3.89 | 7.241 | N | - | - |
| 04225d301 | 93088 | 224.40 | 2.74 | 93088 | 182396 | 10.55 | 0.147 | 27.58 | 9695 | 17827 | 0.052 | 0.07 | 9124 | 16600 | 2.88 | 2.320 | Y | 2.90 | 0.703 |
| 04225d307 | 105188 | 241.04 | 3.09 | 105188 | 205502 | 12.31 | 0.150 | 32.31 | 11011 | 20030 | 0.055 | 0.08 | 9208 | 16638 | 2.90 | 2.489 | N | - | - |
| 04225d297 | 118827 | 235.34 | 2.79 | 118827 | 233408 | 13.26 | 0.146 | 36.46 | 12305 | 22901 | 0.046 | 0.09 | 11317 | 20965 | 4.37 | 4.985 | N | - | - |
| 04225d305 | 138267 | 429.64 | 2.83 | 138267 | 272114 | 15.39 | 0.182 | 44.68 | 14289 | 26726 | 0.063 | 0.11 | 13136 | 23789 | 4.19 | 5.718 | N | - | - |
| 04298d69 | 66001 | 144.09 | 2.85 | 66001 | 129856 | 6.63 | 0.153 | 19.82 | 6807 | 12766 | 0.053 | 0.05 | 6156 | 11459 | 1.67 | 2.862 | Y | 2.85 | 0.724 |
| 04225d299 | 106175 | 226.81 | 2.79 | 106175 | 208788 | 11.98 | 0.158 | 31.30 | 10962 | 20517 | 0.050 | 0.08 | 9357 | 17051 | 2.76 | 2.261 | Y | 2.84 | 0.552 |
| 04203d436 | 102351 | 275.94 | 3.67 | 102351 | 201690 | 9.55 | 0.210 | 29.72 | 10542 | 19847 | 0.065 | 0.08 | 10107 | 18883 | 3.25 | 5.552 | N | - | - |
| 04233d400 | 126121 | 229.45 | 2.68 | 126121 | 248574 | 12.13 | 0.141 | 38.17 | 12974 | 24458 | 0.051 | 0.09 | 10692 | 20063 | 3.38 | 3.964 | Y | 3.05 | 0.572 |
| 04320d272 | 68614 | 158.19 | 2.83 | 68614 | 134330 | 7.12 | 0.155 | 19.14 | 7175 | 13097 | 0.051 | 0.05 | 6424 | 11648 | 1.83 | 3.276 | Y | 3.03 | 1.174 |
| 04320d270 | 117107 | 230.80 | 2.88 | 117107 | 229344 | 12.75 | 0.147 | 33.46 | 12190 | 22443 | 0.048 | 0.09 | 10830 | 20084 | 5.06 | 7.584 | N | - | - |
| 04203d438 | 130495 | 226.82 | 3.26 | 130495 | 257132 | 12.11 | 0.136 | 37.93 | 13437 | 25316 | 0.040 | 0.11 | 12709 | 23805 | 3.59 | 4.988 | N | - | - |
| 04217d399 | 89930 | 180.42 | 3.54 | 89930 | 174198 | 8.54 | 0.141 | 26.24 | 9497 | 16901 | 0.054 | 0.06 | 5884 | 10910 | 1.89 | 3.822 | Y | 2.79 | 1.352 |
| 04320d274 | 115574 | 236.19 | 2.87 | 115574 | 227114 | 12.34 | 0.146 | 32.75 | 11978 | 22272 | 0.044 | 0.09 | 11307 | 20922 | 5.11 | 7.564 | N | - | - |
| 04301d244 | 113446 | 214.58 | 2.71 | 113446 | 223526 | 13.54 | 0.142 | 34.91 | 11672 | 22002 | 0.048 | 0.09 | 10877 | 20253 | 2.50 | 2.623 | Y | 3.05 | 0.428 |
| 04287d53 | 112681 | 234.50 | 2.82 | 112681 | 220586 | 10.89 | 0.145 | 32.94 | 11746 | 21534 | 0.048 | 0.08 | 6847 | 12663 | 1.83 | 3.064 | N | - | - |
| 04237d149 | 76811 | 164.86 | 2.73 | 76811 | 150466 | 6.90 | 0.152 | 23.29 | 7988 | 14707 | 0.051 | 0.06 | 6270 | 11635 | 1.92 | 2.653 | Y | 2.75 | 0.931 |
| 04301d242 | 105427 | 179.56 | 2.39 | 105427 | 207860 | 12.74 | 0.131 | 31.87 | 10820 | 20494 | 0.046 | 0.08 | 9817 | 18440 | 2.15 | 2.649 | Y | 2.95 | 0.509 |
| 04309d171 | 119184 | 215.46 | 3.04 | 119184 | 232354 | 11.84 | 0.145 | 34.63 | 12542 | 22602 | 0.060 | 0.09 | 8467 | 15521 | 2.28 | 2.827 | N | - | - |
| 04237d145 | 110714 | 214.54 | 2.80 | 110714 | 217388 | 10.23 | 0.157 | 36.06 | 11458 | 21325 | 0.054 | 0.08 | 10559 | 19167 | 3.33 | 2.430 | Y | 2.95 | 0.485 |
| 04237d141 | 81691 | 164.70 | 2.76 | 81691 | 160302 | 7.40 | 0.150 | 23.70 | 8467 | 15707 | 0.052 | 0.06 | 7351 | 13482 | 2.04 | 2.631 | Y | 2.78 | 0.846 |
| 04319d198 | 115118 | 232.59 | 2.85 | 115118 | 225798 | 12.61 | 0.155 | 31.79 | 11927 | 22123 | 0.054 | 0.09 | 10995 | 20056 | 3.03 | 2.904 | Y | 3.18 | 0.867 |
| 04237d143 | 75750 | 157.58 | 2.78 | 75750 | 149022 | 7.17 | 0.162 | 23.90 | 7799 | 14653 | 0.056 | 0.05 | 7103 | 12997 | 1.95 | 2.564 | Y | 2.76 | 0.971 |
| 04265d261 | 101482 | 262.51 | 2.87 | 101482 | 199622 | 9.99 | 0.197 | 30.09 | 10491 | 19601 | 0.060 | 0.07 | 5582 | 10215 | 3.04 | 3.383 | N | - | - |

Table C.1: Results of 200 tests on the FRGC dataset

| Range Scan | | | Triangulation | | | Smoothing | | Decimation | | | | Cleanup | | | Registration | | | Deformation | |
|------------|--------|------------|---------------|--------|--------|-----------|------------------|------------|-------|-------|------------------|---------|-------|-------|--------------|------------------|---|-------------|------------------|
| ID | N_p | r_{\max} | t | N_v | N_t | t | e_{rms} | t | N_v | N_t | e_{rms} | t | N_v | N_t | t | e_{rms} | R | t | e_{rms} |
| 04265d263 | 110454 | 219.64 | 2.88 | 110454 | 216326 | 11.11 | 0.137 | 33.10 | 11473 | 21184 | 0.044 | 0.08 | 6300 | 11706 | 1.79 | 2.861 | N | - | - |
| 04265d265 | 83096 | 155.40 | 2.88 | 83096 | 163490 | 8.04 | 0.145 | 24.65 | 8594 | 16053 | 0.049 | 0.06 | 6576 | 12152 | 1.88 | 2.568 | Y | 2.80 | 1.013 |
| 02463d548 | 109732 | 201.20 | 2.37 | 109732 | 216076 | 8.57 | 0.156 | 30.29 | 11314 | 21237 | 0.048 | 0.09 | 9608 | 18021 | 1.85 | 2.670 | Y | 2.87 | 0.557 |
| 02463d546 | 81419 | 231.76 | 2.34 | 81419 | 160052 | 6.30 | 0.211 | 19.88 | 8437 | 15693 | 0.069 | 0.07 | 7726 | 14186 | 2.13 | 2.798 | Y | 2.90 | 0.836 |
| 04222d393 | 102903 | 201.87 | 2.81 | 102903 | 202558 | 10.29 | 0.148 | 32.29 | 10617 | 19907 | 0.047 | 0.08 | 10037 | 18508 | 2.75 | 2.747 | Y | 2.95 | 0.645 |
| 04222d391 | 107941 | 192.90 | 2.81 | 107941 | 212972 | 11.89 | 0.136 | 32.90 | 11088 | 20988 | 0.046 | 0.08 | 10616 | 19774 | 2.70 | 2.712 | Y | 2.91 | 0.621 |
| 04320d278 | 119818 | 240.23 | 2.92 | 119818 | 235356 | 13.17 | 0.163 | 33.98 | 12403 | 23076 | 0.054 | 0.09 | 11356 | 20711 | 4.14 | 6.744 | N | - | - |
| 04222d397 | 114391 | 209.78 | 2.93 | 114391 | 225110 | 11.89 | 0.149 | 34.28 | 11807 | 22117 | 0.050 | 0.09 | 10554 | 19838 | 3.55 | 3.664 | Y | 2.99 | 0.710 |
| 04222d395 | 115070 | 202.02 | 2.93 | 115070 | 226438 | 12.02 | 0.140 | 34.89 | 11870 | 22257 | 0.045 | 0.09 | 10418 | 19538 | 2.35 | 2.664 | Y | 2.89 | 0.689 |
| 04288d256 | 101889 | 217.24 | 2.79 | 101889 | 199170 | 9.92 | 0.151 | 30.01 | 10587 | 19517 | 0.060 | 0.08 | 8346 | 15517 | 2.04 | 2.887 | Y | 2.90 | 0.494 |
| 04321d108 | 74185 | 221.83 | 2.79 | 74185 | 144768 | 7.79 | 0.220 | 20.87 | 7803 | 14046 | 0.087 | 0.05 | 7160 | 12275 | 2.40 | 2.437 | Y | 2.95 | 0.659 |
| 04309d163 | 100040 | 180.53 | 2.79 | 100040 | 195888 | 10.02 | 0.142 | 28.26 | 10450 | 19136 | 0.056 | 0.08 | 9084 | 16678 | 2.35 | 2.890 | Y | 3.46 | 0.628 |
| 04301d256 | 115698 | 218.00 | 2.83 | 115698 | 228176 | 12.17 | 0.145 | 34.77 | 11905 | 22472 | 0.047 | 0.09 | 11103 | 20707 | 3.30 | 2.962 | Y | 3.16 | 0.586 |
| 04301d254 | 104635 | 207.77 | 2.91 | 104635 | 206284 | 11.08 | 0.150 | 30.21 | 10764 | 20318 | 0.052 | 0.08 | 9919 | 18235 | 2.58 | 2.622 | Y | 3.08 | 0.627 |
| 04301d252 | 100642 | 181.88 | 2.95 | 100642 | 197620 | 11.89 | 0.143 | 29.12 | 10363 | 19434 | 0.055 | 0.08 | 9301 | 17203 | 2.14 | 2.653 | Y | 2.94 | 0.481 |
| 04301d250 | 106899 | 182.86 | 2.76 | 106899 | 210876 | 12.72 | 0.133 | 33.93 | 10977 | 20782 | 0.050 | 0.08 | 10081 | 18727 | 2.17 | 2.640 | Y | 2.98 | 0.487 |
| 04301d258 | 97068 | 171.75 | 2.79 | 97068 | 191492 | 9.75 | 0.137 | 28.02 | 9972 | 18864 | 0.044 | 0.07 | 9472 | 17807 | 2.26 | 2.769 | Y | 2.97 | 0.438 |
| 04226d361 | 82287 | 163.70 | 2.75 | 82287 | 160816 | 8.09 | 0.143 | 25.36 | 8567 | 15733 | 0.050 | 0.06 | 7169 | 13316 | 2.18 | 3.264 | Y | 2.80 | 1.129 |
| 04237d139 | 72837 | 222.75 | 2.76 | 72837 | 142936 | 7.11 | 0.206 | 19.90 | 7562 | 13992 | 0.066 | 0.05 | 7119 | 12938 | 3.06 | 2.594 | Y | 2.82 | 0.780 |
| 04311d236 | 90015 | 154.55 | 2.94 | 90015 | 176912 | 8.87 | 0.123 | 25.68 | 9326 | 17360 | 0.039 | 0.07 | 8262 | 15461 | 2.14 | 3.970 | Y | 3.06 | 2.299 |
| 04311d234 | 85701 | 176.08 | 3.30 | 85701 | 167554 | 8.38 | 0.147 | 23.77 | 8928 | 16364 | 0.052 | 0.06 | 7649 | 14045 | 1.89 | 3.107 | Y | 2.93 | 1.571 |
| 04311d232 | 86839 | 175.71 | 3.18 | 86839 | 170262 | 8.69 | 0.151 | 23.81 | 9026 | 16671 | 0.049 | 0.07 | 8213 | 15045 | 2.21 | 2.533 | Y | 2.98 | 1.016 |
| 04329d112 | 109795 | 219.35 | 2.94 | 109795 | 214024 | 10.89 | 0.151 | 29.58 | 11516 | 20858 | 0.058 | 0.08 | 9349 | 17201 | 2.72 | 2.430 | N | - | - |
| 04311d230 | 91793 | 159.92 | 2.97 | 91793 | 180742 | 9.15 | 0.148 | 27.14 | 9496 | 17737 | 0.051 | 0.07 | 8354 | 15186 | 2.07 | 2.859 | Y | 2.96 | 1.445 |
| 04309d173 | 84790 | 192.68 | 2.94 | 84790 | 164048 | 8.22 | 0.143 | 24.99 | 8980 | 15870 | 0.060 | 0.06 | 6860 | 12765 | 1.72 | 2.449 | Y | 2.95 | 0.821 |
| 04261d301 | 105849 | 211.95 | 2.93 | 105849 | 207494 | 10.36 | 0.144 | 31.53 | 11044 | 20274 | 0.046 | 0.08 | 8503 | 16016 | 2.34 | 3.322 | Y | 2.87 | 1.216 |
| 04327d300 | 107397 | 222.93 | 2.94 | 107397 | 208334 | 10.36 | 0.189 | 29.25 | 11366 | 20156 | 0.108 | 0.07 | 9845 | 16697 | 2.34 | 2.903 | Y | 3.18 | 1.212 |
| 04309d175 | 95502 | 177.30 | 2.96 | 95502 | 186570 | 9.11 | 0.130 | 27.84 | 9996 | 18214 | 0.049 | 0.07 | 8388 | 15707 | 2.01 | 2.568 | Y | 3.03 | 0.572 |
| 02463d550 | 104425 | 190.07 | 2.32 | 104425 | 205176 | 7.91 | 0.157 | 31.83 | 10832 | 20109 | 0.051 | 0.08 | 9542 | 17861 | 1.87 | 2.677 | Y | 2.87 | 0.672 |
| 04329d100 | 101925 | 213.68 | 2.93 | 101925 | 199060 | 9.99 | 0.151 | 28.81 | 10646 | 19445 | 0.060 | 0.07 | 9247 | 16800 | 2.80 | 1.982 | Y | 3.13 | 0.442 |
| 02463d554 | 102587 | 184.45 | 2.36 | 102587 | 202176 | 7.83 | 0.160 | 33.97 | 10581 | 19879 | 0.054 | 0.08 | 9342 | 17391 | 1.87 | 3.284 | Y | 2.89 | 0.816 |
| 02463d556 | 108302 | 202.28 | 2.35 | 108302 | 213566 | 8.41 | 0.158 | 34.02 | 11149 | 21017 | 0.056 | 0.08 | 9921 | 18261 | 1.99 | 2.742 | Y | 2.93 | 0.870 |

Table C.1: Results of 200 tests on the FRGC dataset

| Range Scan | | | Triangulation | | | Smoothing | | Decimation | | | | Cleanup | | | Registration | | | Deformation | |
|------------|--------|------------|---------------|--------|--------|-----------|------------------|------------|-------|-------|------------------|---------|-------|-------|--------------|------------------|---|-------------|------------------|
| ID | N_p | r_{\max} | t | N_v | N_t | t | e_{rms} | t | N_v | N_t | e_{rms} | t | N_v | N_t | t | e_{rms} | R | t | e_{rms} |
| 02463d558 | 104413 | 220.30 | 2.39 | 104413 | 205492 | 8.90 | 0.168 | 31.05 | 10775 | 20197 | 0.057 | 0.07 | 8785 | 16139 | 1.79 | 3.135 | Y | 2.84 | 1.094 |
| 04286d267 | 98799 | 217.07 | 2.92 | 98799 | 192548 | 9.44 | 0.143 | 28.77 | 10289 | 18823 | 0.056 | 0.07 | 8599 | 15553 | 2.52 | 2.779 | Y | 2.93 | 0.692 |
| 04239d390 | 106779 | 230.26 | 2.94 | 106779 | 209298 | 10.11 | 0.144 | 32.31 | 11125 | 20462 | 0.050 | 0.08 | 10273 | 18761 | 3.62 | 3.725 | Y | 3.14 | 1.096 |
| 04286d263 | 106797 | 219.37 | 2.76 | 106797 | 209234 | 10.39 | 0.140 | 31.17 | 11037 | 20544 | 0.052 | 0.08 | 9848 | 17763 | 3.26 | 2.701 | Y | 2.94 | 0.613 |
| 04320d276 | 72015 | 166.14 | 2.99 | 72015 | 140658 | 7.32 | 0.152 | 19.96 | 7517 | 13734 | 0.052 | 0.05 | 6752 | 12321 | 2.08 | 3.703 | Y | 3.03 | 1.284 |
| 04288d260 | 81635 | 157.53 | 2.78 | 81635 | 160404 | 8.17 | 0.146 | 24.51 | 8423 | 15765 | 0.049 | 0.06 | 7911 | 14669 | 2.09 | 3.447 | Y | 2.91 | 1.085 |
| 04311d228 | 99086 | 179.86 | 3.00 | 99086 | 194284 | 10.34 | 0.140 | 29.20 | 10291 | 19031 | 0.047 | 0.08 | 9342 | 17114 | 2.66 | 2.523 | Y | 3.04 | 1.140 |
| 04221d433 | 91857 | 192.09 | 2.87 | 91857 | 180736 | 10.19 | 0.148 | 27.50 | 9505 | 17739 | 0.045 | 0.07 | 9129 | 16960 | 2.70 | 3.366 | Y | 2.94 | 0.873 |
| 04226d357 | 64034 | 217.59 | 2.84 | 64034 | 125422 | 6.66 | 0.196 | 19.20 | 6675 | 12261 | 0.065 | 0.04 | 4140 | 7588 | 1.72 | 3.177 | Y | 2.70 | 0.868 |
| 04309d167 | 87204 | 182.27 | 2.78 | 87204 | 171420 | 8.92 | 0.151 | 20.87 | 9035 | 16819 | 0.054 | 0.07 | 8612 | 15729 | 2.36 | 2.848 | Y | 3.05 | 0.707 |
| 04221d437 | 108157 | 192.66 | 2.89 | 108157 | 212980 | 11.87 | 0.144 | 32.38 | 11162 | 20926 | 0.044 | 0.09 | 10525 | 19561 | 3.14 | 3.327 | Y | 2.99 | 1.000 |
| 04221d439 | 101484 | 170.35 | 2.80 | 101484 | 198928 | 11.37 | 0.136 | 30.99 | 10562 | 19463 | 0.043 | 0.08 | 9668 | 17805 | 2.81 | 3.873 | Y | 2.92 | 1.362 |
| 04302d116 | 82846 | 165.48 | 2.87 | 82846 | 162574 | 8.45 | 0.141 | 24.28 | 8605 | 15916 | 0.056 | 0.06 | 7935 | 14588 | 1.90 | 2.233 | Y | 2.92 | 0.497 |
| 04226d359 | 80028 | 175.30 | 2.94 | 80028 | 156980 | 8.01 | 0.138 | 24.50 | 8303 | 15389 | 0.043 | 0.06 | 6798 | 12829 | 2.03 | 3.469 | Y | 2.79 | 1.660 |
| 04311d226 | 86749 | 185.94 | 3.01 | 86749 | 169408 | 8.77 | 0.142 | 24.72 | 9087 | 16513 | 0.051 | 0.07 | 8406 | 15042 | 3.11 | 4.202 | N | - | - |
| 04321d112 | 76819 | 171.74 | 2.81 | 76819 | 149808 | 7.62 | 0.153 | 20.97 | 8061 | 14578 | 0.059 | 0.06 | 7331 | 13127 | 1.95 | 2.415 | Y | 2.99 | 0.630 |
| 04309d169 | 100505 | 203.13 | 2.96 | 100505 | 196846 | 10.40 | 0.141 | 27.32 | 10512 | 19201 | 0.052 | 0.08 | 10022 | 18273 | 2.70 | 2.805 | Y | 3.17 | 0.940 |
| 04309d165 | 82968 | 177.46 | 2.82 | 82968 | 162476 | 8.57 | 0.154 | 20.32 | 8646 | 15888 | 0.057 | 0.06 | 8253 | 15146 | 2.27 | 2.830 | Y | 3.11 | 0.647 |
| 04298d71 | 96758 | 200.37 | 3.08 | 96758 | 189792 | 9.53 | 0.149 | 28.22 | 10025 | 18625 | 0.057 | 0.07 | 8647 | 15938 | 2.59 | 2.680 | Y | 3.00 | 0.520 |
| 04309d161 | 116890 | 239.69 | 2.97 | 116890 | 228860 | 11.67 | 0.172 | 34.93 | 12162 | 22390 | 0.061 | 0.09 | 11491 | 21041 | 3.63 | 4.452 | N | - | - |
| 04301d240 | 99694 | 268.38 | 2.83 | 99694 | 196262 | 11.85 | 0.207 | 28.92 | 10342 | 19244 | 0.063 | 0.08 | 9793 | 18221 | 4.00 | 4.091 | N | - | - |
| 04239d388 | 107553 | 234.83 | 2.94 | 107553 | 210702 | 10.38 | 0.141 | 32.26 | 11167 | 20632 | 0.050 | 0.07 | 7225 | 13255 | 2.23 | 3.406 | Y | 2.83 | 0.846 |
| 04239d384 | 109115 | 221.23 | 2.79 | 109115 | 213856 | 10.52 | 0.145 | 33.47 | 11375 | 20889 | 0.052 | 0.08 | 7205 | 13335 | 2.15 | 3.103 | Y | 2.83 | 1.186 |
| 04286d277 | 85097 | 159.55 | 2.85 | 85097 | 166222 | 8.13 | 0.132 | 24.33 | 8869 | 16234 | 0.048 | 0.06 | 6886 | 12815 | 1.84 | 2.639 | Y | 2.85 | 0.825 |
| 04239d386 | 100479 | 228.00 | 2.76 | 100479 | 197308 | 9.68 | 0.150 | 29.91 | 10395 | 19359 | 0.052 | 0.07 | 6293 | 11638 | 2.13 | 3.280 | Y | 2.78 | 0.520 |
| 04286d275 | 88812 | 169.72 | 3.10 | 88812 | 174406 | 8.56 | 0.145 | 26.19 | 9146 | 17159 | 0.056 | 0.07 | 8220 | 14966 | 2.36 | 2.579 | Y | 2.86 | 0.683 |
| 04239d380 | 69462 | 166.12 | 2.74 | 69462 | 135974 | 6.76 | 0.160 | 24.48 | 7244 | 13283 | 0.062 | 0.05 | 5913 | 10953 | 1.93 | 3.164 | Y | 2.76 | 0.809 |
| 04286d273 | 84929 | 185.26 | 2.81 | 84929 | 165428 | 8.33 | 0.141 | 25.37 | 8907 | 16103 | 0.053 | 0.06 | 6019 | 11100 | 1.63 | 2.474 | Y | 2.75 | 0.842 |
| 04239d382 | 92914 | 214.14 | 2.79 | 92914 | 181942 | 9.24 | 0.154 | 32.47 | 9690 | 17778 | 0.051 | 0.07 | 6192 | 11475 | 2.03 | 3.213 | Y | 2.79 | 0.657 |
| 04286d271 | 137162 | 242.11 | 3.24 | 137162 | 268330 | 13.41 | 0.158 | 40.94 | 14202 | 26273 | 0.054 | 0.10 | 7323 | 12826 | 2.61 | 2.202 | N | - | - |
| 04221d445 | 82673 | 157.09 | 2.82 | 82673 | 162642 | 8.87 | 0.142 | 24.48 | 8567 | 15951 | 0.045 | 0.06 | 8142 | 15058 | 2.17 | 3.203 | Y | 2.85 | 0.894 |
| 04221d441 | 119120 | 232.05 | 3.12 | 119120 | 234208 | 13.02 | 0.157 | 36.77 | 12323 | 22992 | 0.059 | 0.09 | 11224 | 20317 | 4.25 | 3.683 | N | - | - |

Table C.1: Results of 200 tests on the FRGC dataset

| Range Scan | | | Triangulation | | | Smoothing | | Decimation | | | | Cleanup | | | Registration | | | Deformation | |
|------------|--------|------------|---------------|--------|--------|-----------|------------------|------------|-------|-------|------------------|---------|-------|-------|--------------|------------------|---|-------------|------------------|
| ID | N_p | r_{\max} | t | N_v | N_t | t | e_{rms} | t | N_v | N_t | e_{rms} | t | N_v | N_t | t | e_{rms} | R | t | e_{rms} |
| 04221d443 | 118239 | 232.67 | 2.72 | 118239 | 232706 | 13.34 | 0.159 | 35.85 | 12182 | 22893 | 0.045 | 0.09 | 11276 | 21075 | 4.14 | 5.977 | N | - | - |
| 04213d280 | 120792 | 188.27 | 3.55 | 120792 | 234534 | 11.34 | 0.172 | 35.52 | 12794 | 22697 | 0.083 | 0.09 | 11757 | 20333 | 1.96 | 1.993 | Y | 2.82 | 0.917 |
| 04334d310 | 82264 | 179.41 | 2.90 | 82264 | 160176 | 8.20 | 0.167 | 22.81 | 8616 | 15608 | 0.066 | 0.06 | 5729 | 10578 | 1.61 | 2.877 | Y | 3.02 | 0.840 |
| 04327d292 | 114502 | 206.92 | 2.80 | 114502 | 221390 | 11.10 | 0.182 | 32.37 | 12246 | 21293 | 0.094 | 0.08 | 9992 | 17269 | 2.13 | 3.256 | Y | 3.03 | 1.840 |
| 04221d429 | 78338 | 237.99 | 2.95 | 78338 | 154104 | 8.68 | 0.196 | 23.28 | 8126 | 15103 | 0.057 | 0.06 | 7974 | 14604 | 3.53 | 6.000 | N | - | - |
| 04298d67 | 66185 | 153.83 | 2.84 | 66185 | 129956 | 6.89 | 0.169 | 19.45 | 6842 | 12762 | 0.056 | 0.05 | 6090 | 11288 | 1.74 | 2.749 | Y | 2.82 | 0.613 |
| 04273d246 | 87763 | 167.84 | 2.87 | 87763 | 172838 | 8.41 | 0.141 | 25.89 | 9035 | 17017 | 0.045 | 0.07 | 8469 | 15880 | 2.00 | 2.705 | Y | 2.80 | 0.602 |
| 04329d106 | 77087 | 177.35 | 2.87 | 77087 | 149486 | 7.66 | 0.145 | 18.90 | 8131 | 14504 | 0.065 | 0.06 | 6891 | 12426 | 1.81 | 2.263 | Y | 3.03 | 0.805 |
| 04286d269 | 142134 | 243.43 | 2.99 | 142134 | 279730 | 13.75 | 0.146 | 46.25 | 14660 | 27512 | 0.043 | 0.11 | 7427 | 13935 | 2.72 | 2.153 | N | - | - |
| 04329d104 | 100045 | 201.75 | 2.96 | 100045 | 195318 | 9.82 | 0.149 | 22.66 | 10419 | 19077 | 0.061 | 0.08 | 9818 | 17647 | 2.58 | 1.907 | Y | 3.13 | 0.490 |
| 04201d368 | 103061 | 193.98 | 2.66 | 103061 | 202160 | 9.22 | 0.147 | 35.52 | 10666 | 19814 | 0.053 | 0.08 | 9661 | 17734 | 2.03 | 2.443 | Y | 2.93 | 0.650 |
| 04273d248 | 101283 | 216.26 | 3.04 | 101283 | 198352 | 9.62 | 0.157 | 30.40 | 10540 | 19431 | 0.058 | 0.08 | 9533 | 17599 | 2.82 | 2.855 | Y | 2.94 | 0.679 |
| 04225d291 | 111931 | 352.88 | 2.86 | 111931 | 220708 | 12.25 | 0.226 | 32.54 | 11548 | 21693 | 0.076 | 0.09 | 11246 | 20376 | 4.33 | 6.025 | N | - | - |
| 04225d293 | 114801 | 237.44 | 2.74 | 114801 | 225434 | 13.20 | 0.155 | 34.33 | 11893 | 22100 | 0.054 | 0.08 | 10962 | 19705 | 3.73 | 2.399 | N | - | - |
| 04225d295 | 141846 | 243.72 | 2.86 | 141846 | 278732 | 16.06 | 0.150 | 42.87 | 14687 | 27342 | 0.047 | 0.11 | 13564 | 24728 | 5.02 | 6.097 | N | - | - |
| 04236d158 | 84678 | 171.35 | 2.79 | 84678 | 165534 | 8.04 | 0.154 | 22.85 | 8858 | 16136 | 0.058 | 0.07 | 8231 | 14918 | 2.27 | 3.005 | Y | 2.83 | 0.805 |
| 04410d182 | 93771 | 174.35 | 2.34 | 93771 | 184304 | 7.09 | 0.141 | 21.64 | 9722 | 18076 | 0.046 | 0.07 | 8115 | 15346 | 1.93 | 2.622 | Y | 3.02 | 0.597 |
| 04236d156 | 98398 | 193.06 | 2.77 | 98398 | 192476 | 9.86 | 0.146 | 23.11 | 10255 | 18794 | 0.063 | 0.07 | 8884 | 16227 | 2.66 | 3.454 | Y | 2.92 | 0.763 |
| 04239d378 | 81410 | 155.99 | 2.82 | 81410 | 159238 | 8.17 | 0.140 | 29.05 | 8484 | 15577 | 0.049 | 0.06 | 7265 | 13444 | 2.40 | 3.431 | Y | 2.87 | 1.208 |
| 04233d402 | 126731 | 194.53 | 2.84 | 126731 | 250206 | 13.91 | 0.128 | 37.16 | 13073 | 24623 | 0.050 | 0.09 | 11500 | 21533 | 3.28 | 3.979 | Y | 3.11 | 0.774 |
| 04300d228 | 86539 | 178.35 | 2.78 | 86539 | 169602 | 10.28 | 0.150 | 26.01 | 9026 | 16580 | 0.053 | 0.06 | 7562 | 14050 | 1.95 | 2.360 | Y | 2.95 | 0.701 |
| 04286d265 | 73843 | 177.89 | 2.78 | 73843 | 144280 | 6.85 | 0.156 | 21.32 | 7672 | 14121 | 0.057 | 0.05 | 6617 | 11887 | 2.12 | 2.775 | Y | 2.83 | 0.688 |
| 04219d419 | 97506 | 183.45 | 2.89 | 97506 | 191004 | 11.22 | 0.151 | 29.16 | 10191 | 18651 | 0.055 | 0.07 | 8423 | 15610 | 2.41 | 2.867 | Y | 2.82 | 0.951 |
| 04219d417 | 78825 | 178.54 | 2.96 | 78825 | 154244 | 8.82 | 0.140 | 23.27 | 8233 | 15061 | 0.046 | 0.06 | 7370 | 13491 | 2.15 | 3.246 | Y | 2.85 | 1.358 |
| 04219d415 | 72553 | 167.70 | 3.09 | 72553 | 140894 | 7.88 | 0.143 | 20.99 | 7650 | 13687 | 0.054 | 0.05 | 6454 | 11791 | 2.07 | 4.037 | Y | 2.80 | 2.174 |
| 04327d298 | 114079 | 233.45 | 2.73 | 114079 | 218884 | 11.29 | 0.175 | 31.39 | 12293 | 20965 | 0.088 | 0.08 | 8670 | 15258 | 2.44 | 3.180 | Y | 3.21 | 1.302 |
| 04273d250 | 95562 | 200.69 | 2.92 | 95562 | 187120 | 9.12 | 0.155 | 28.78 | 9961 | 18307 | 0.056 | 0.07 | 9153 | 16942 | 2.53 | 3.129 | Y | 2.91 | 0.834 |
| 04273d254 | 87881 | 181.16 | 2.80 | 87881 | 172154 | 8.85 | 0.155 | 25.48 | 9170 | 16836 | 0.059 | 0.07 | 8238 | 15202 | 2.39 | 3.630 | Y | 2.99 | 1.203 |
| 04273d256 | 104686 | 207.18 | 2.86 | 104686 | 205612 | 10.47 | 0.141 | 31.28 | 10847 | 20186 | 0.049 | 0.08 | 10317 | 19017 | 2.88 | 2.873 | Y | 3.05 | 0.700 |
| 04201d374 | 107657 | 209.72 | 3.04 | 107657 | 209400 | 9.92 | 0.142 | 36.50 | 11307 | 20405 | 0.059 | 0.08 | 8823 | 16183 | 1.82 | 2.512 | Y | 2.86 | 0.634 |
| 04319d192 | 117144 | 185.83 | 2.91 | 117144 | 229584 | 12.29 | 0.138 | 32.30 | 12129 | 22499 | 0.050 | 0.09 | 10769 | 19718 | 2.07 | 2.462 | Y | 3.09 | 0.967 |
| 04201d376 | 111938 | 226.11 | 2.87 | 111938 | 219956 | 10.51 | 0.157 | 37.32 | 11557 | 21599 | 0.054 | 0.08 | 10599 | 19413 | 2.62 | 2.450 | Y | 3.05 | 0.841 |

Table C.1: Results of 200 tests on the FRGC dataset

| Range Scan | | | Triangulation | | | Smoothing | | Decimation | | | | Cleanup | | | Registration | | | Deformation | |
|------------|--------|------------|---------------|--------|--------|-----------|------------------|------------|-------|-------|------------------|---------|-------|-------|--------------|------------------|---|-------------|------------------|
| ID | N_p | r_{\max} | t | N_v | N_t | t | e_{rms} | t | N_v | N_t | e_{rms} | t | N_v | N_t | t | e_{rms} | R | t | e_{rms} |
| 04324d290 | 125573 | 237.67 | 2.89 | 125573 | 246020 | 12.74 | 0.154 | 35.16 | 13068 | 24054 | 0.054 | 0.09 | 7616 | 13843 | 2.17 | 2.887 | N | - | - |
| 04201d372 | 115681 | 186.84 | 2.85 | 115681 | 226466 | 10.54 | 0.144 | 38.59 | 11991 | 22194 | 0.057 | 0.09 | 10737 | 19607 | 2.10 | 2.397 | Y | 2.91 | 0.443 |
| 04300d222 | 101734 | 198.39 | 2.95 | 101734 | 198892 | 11.91 | 0.142 | 29.36 | 10636 | 19414 | 0.054 | 0.07 | 7653 | 14334 | 1.87 | 2.363 | Y | 2.94 | 0.622 |
| 04300d226 | 85372 | 196.56 | 2.99 | 85372 | 166334 | 10.05 | 0.152 | 24.41 | 9009 | 16169 | 0.063 | 0.06 | 6713 | 12580 | 1.67 | 2.172 | Y | 2.87 | 0.559 |
| 04319d194 | 82030 | 181.37 | 2.85 | 82030 | 160132 | 8.03 | 0.171 | 22.85 | 8583 | 15622 | 0.061 | 0.06 | 7424 | 13526 | 1.64 | 2.883 | Y | 2.99 | 1.062 |
| 04287d49 | 127622 | 233.28 | 2.89 | 127622 | 246600 | 11.91 | 0.148 | 38.50 | 13516 | 23791 | 0.058 | 0.10 | 10597 | 18723 | 3.46 | 4.468 | N | - | - |
| 04319d196 | 103792 | 201.87 | 2.92 | 103792 | 202950 | 10.69 | 0.155 | 29.46 | 10812 | 19839 | 0.055 | 0.08 | 8777 | 16283 | 2.08 | 3.632 | Y | 3.06 | 1.382 |
| 04287d47 | 71887 | 162.08 | 2.74 | 71887 | 140732 | 6.99 | 0.159 | 23.18 | 7490 | 13751 | 0.054 | 0.05 | 6907 | 12442 | 2.04 | 3.110 | Y | 2.83 | 1.153 |
| 04287d45 | 83243 | 178.44 | 2.95 | 83243 | 162786 | 8.23 | 0.143 | 24.49 | 8727 | 15856 | 0.048 | 0.06 | 6754 | 12512 | 1.87 | 3.114 | Y | 2.86 | 0.886 |
| 04279d289 | 110096 | 214.45 | 2.81 | 110096 | 215214 | 10.94 | 0.167 | 32.97 | 11510 | 20994 | 0.067 | 0.07 | 6359 | 11670 | 1.77 | 2.212 | Y | 2.80 | 0.834 |
| 04297d261 | 92453 | 175.59 | 2.90 | 92453 | 180440 | 9.14 | 0.140 | 27.90 | 9689 | 17588 | 0.049 | 0.07 | 8762 | 16101 | 2.13 | 3.283 | Y | 2.93 | 1.359 |
| 04297d267 | 112295 | 211.63 | 2.97 | 112295 | 217782 | 11.29 | 0.148 | 33.28 | 11733 | 21204 | 0.062 | 0.08 | 9553 | 17520 | 2.40 | 2.992 | Y | 3.00 | 0.789 |
| 04233d398 | 134163 | 241.20 | 2.47 | 134163 | 264570 | 13.20 | 0.141 | 42.33 | 13767 | 26088 | 0.045 | 0.11 | 13434 | 25455 | 4.29 | 5.326 | N | - | - |
| 04297d265 | 106235 | 172.71 | 2.99 | 106235 | 207516 | 10.00 | 0.136 | 30.75 | 11131 | 20207 | 0.055 | 0.08 | 9800 | 17759 | 2.28 | 2.910 | Y | 2.94 | 0.665 |
| 04233d394 | 120609 | 234.68 | 2.74 | 120609 | 238184 | 12.07 | 0.153 | 33.64 | 12394 | 23468 | 0.048 | 0.10 | 12042 | 22744 | 4.09 | 5.144 | N | - | - |
| 04279d283 | 112497 | 192.03 | 2.84 | 112497 | 219790 | 10.80 | 0.170 | 34.17 | 11752 | 21427 | 0.074 | 0.09 | 10154 | 18030 | 2.18 | 2.092 | Y | 2.87 | 0.598 |
| 04233d396 | 113751 | 214.19 | 2.89 | 113751 | 224422 | 11.76 | 0.147 | 34.10 | 11684 | 22114 | 0.048 | 0.09 | 11359 | 21527 | 3.82 | 5.098 | N | - | - |
| 04233d392 | 135914 | 235.68 | 2.92 | 135914 | 268334 | 13.52 | 0.138 | 37.80 | 13924 | 26482 | 0.047 | 0.11 | 13469 | 25572 | 4.09 | 4.285 | Y | 3.19 | 1.707 |
| 04202d440 | 73777 | 160.24 | 2.83 | 73777 | 144952 | 7.05 | 0.156 | 22.29 | 7646 | 14216 | 0.046 | 0.05 | 6127 | 11554 | 1.55 | 3.254 | Y | 2.80 | 1.200 |
| 04202d442 | 82407 | 183.12 | 3.08 | 82407 | 162250 | 8.02 | 0.159 | 24.58 | 8513 | 15934 | 0.050 | 0.06 | 7507 | 13701 | 1.89 | 3.165 | Y | 2.92 | 0.961 |
| 04202d444 | 103766 | 212.18 | 3.12 | 103766 | 202776 | 9.92 | 0.144 | 29.81 | 10814 | 19789 | 0.047 | 0.08 | 9999 | 18287 | 2.61 | 3.129 | Y | 3.03 | 1.051 |
| 04202d446 | 77425 | 146.59 | 3.19 | 77425 | 152506 | 7.55 | 0.144 | 22.57 | 7962 | 15021 | 0.042 | 0.05 | 6615 | 12544 | 1.60 | 3.184 | Y | 2.78 | 1.082 |
| 04202d448 | 89112 | 193.90 | 3.24 | 89112 | 174908 | 8.08 | 0.148 | 25.73 | 9248 | 17126 | 0.051 | 0.07 | 8457 | 15438 | 2.18 | 3.498 | Y | 2.97 | 1.431 |
| 04324d286 | 126545 | 248.68 | 2.75 | 126545 | 248436 | 12.90 | 0.158 | 34.69 | 13116 | 24337 | 0.059 | 0.09 | 7509 | 13572 | 1.98 | 2.522 | N | - | - |
| 04324d282 | 121504 | 247.52 | 2.87 | 121504 | 236866 | 11.90 | 0.170 | 32.74 | 12727 | 23074 | 0.071 | 0.09 | 11473 | 20054 | 3.40 | 3.632 | N | - | - |
| 04324d280 | 96048 | 227.00 | 2.80 | 96048 | 187514 | 9.57 | 0.172 | 25.95 | 10073 | 18250 | 0.069 | 0.07 | 9104 | 16097 | 2.77 | 2.688 | Y | 3.18 | 0.704 |
| 04300d218 | 91621 | 207.80 | 2.33 | 91621 | 178998 | 10.25 | 0.162 | 27.19 | 9574 | 17470 | 0.061 | 0.07 | 6420 | 11936 | 1.74 | 2.333 | Y | 2.86 | 0.628 |
| 04324d288 | 118152 | 251.18 | 2.71 | 118152 | 231522 | 12.42 | 0.164 | 32.07 | 12319 | 22609 | 0.060 | 0.08 | 7036 | 12805 | 1.96 | 2.621 | N | - | - |
| 04319d188 | 105338 | 198.94 | 2.88 | 105338 | 206780 | 10.54 | 0.153 | 30.52 | 10921 | 20284 | 0.052 | 0.08 | 9281 | 17344 | 1.84 | 2.488 | Y | 2.97 | 0.865 |
| 04202d438 | 60544 | 176.68 | 2.97 | 60544 | 118766 | 5.57 | 0.202 | 19.98 | 6282 | 11629 | 0.063 | 0.05 | 5893 | 10860 | 1.68 | 3.225 | Y | 2.78 | 1.101 |
| 04265d267 | 87684 | 164.01 | 2.85 | 87684 | 172010 | 8.97 | 0.144 | 25.79 | 9099 | 16856 | 0.053 | 0.06 | 6792 | 12534 | 2.04 | 2.694 | Y | 2.83 | 1.070 |
| 04217d409 | 89413 | 183.78 | 2.99 | 89413 | 173378 | 9.16 | 0.148 | 25.80 | 9420 | 16847 | 0.058 | 0.06 | 7605 | 13918 | 2.48 | 3.828 | Y | 2.90 | 1.231 |

Table C.1: Results of 200 tests on the FRGC dataset

| Range Scan | | | Triangulation | | | Smoothing | | Decimation | | | | Cleanup | | | Registration | | | Deformation | |
|------------|--------|------------|---------------|--------|--------|-----------|------------------|------------|-------|-------|------------------|---------|-------|-------|--------------|------------------|---|-------------|------------------|
| ID | N_p | r_{\max} | t | N_v | N_t | t | e_{rms} | t | N_v | N_t | e_{rms} | t | N_v | N_t | t | e_{rms} | R | t | e_{rms} |
| 04265d269 | 103916 | 201.93 | 2.97 | 103916 | 202692 | 9.87 | 0.190 | 31.23 | 10968 | 19645 | 0.097 | 0.08 | 9789 | 16737 | 2.50 | 2.909 | Y | 2.86 | 1.062 |
| 04217d407 | 93832 | 175.93 | 3.09 | 93832 | 183658 | 9.65 | 0.145 | 28.12 | 9808 | 17934 | 0.057 | 0.06 | 6456 | 11715 | 1.86 | 3.571 | Y | 2.79 | 1.215 |
| 04321d110 | 80131 | 168.84 | 2.87 | 80131 | 156252 | 8.45 | 0.164 | 22.54 | 8413 | 15184 | 0.068 | 0.06 | 7792 | 13983 | 2.09 | 2.468 | Y | 3.00 | 0.786 |
| 04217d405 | 112089 | 221.21 | 3.14 | 112089 | 217898 | 10.48 | 0.165 | 33.35 | 11837 | 21165 | 0.057 | 0.08 | 5491 | 10177 | 1.65 | 3.593 | N | - | - |
| 04217d403 | 103867 | 204.86 | 3.58 | 103867 | 200886 | 9.45 | 0.142 | 33.32 | 11051 | 19415 | 0.056 | 0.07 | 6078 | 11317 | 1.98 | 3.706 | Y | 2.78 | 1.230 |
| 04203d452 | 90245 | 168.21 | 3.58 | 90245 | 177648 | 8.85 | 0.138 | 26.50 | 9310 | 17472 | 0.042 | 0.07 | 8115 | 15288 | 1.68 | 2.470 | Y | 2.81 | 0.990 |
| 04217d401 | 82895 | 168.96 | 3.35 | 82895 | 161582 | 7.74 | 0.137 | 24.71 | 8709 | 15742 | 0.050 | 0.06 | 6133 | 11505 | 2.24 | 4.707 | Y | 2.83 | 1.044 |
| 04203d450 | 122050 | 250.88 | 3.68 | 122050 | 240728 | 11.90 | 0.153 | 36.28 | 12578 | 23695 | 0.044 | 0.09 | 11828 | 22165 | 3.72 | 5.471 | N | - | - |

ID: Range scan file ID

N_p : Number of points

N_v : Number of vertices

N_t : Number of triangles

t : Computation time in seconds

r_{\max} : Radius of bounding box, i.e., $r_{\max} = \sqrt{(x_{\max} - x_{\min})^2 + (y_{\max} - y_{\min})^2 + (z_{\max} - z_{\min})^2} / 2$

e_{rms} : Root-mean-squared error, i.e., $e_{\text{rms}} = \sqrt{(d_1^2 + d_2^2 + \dots + d_n^2) / n}$

R: Results of registration (Y: successful; N: failed)

This page is intentionally left blank.

Table C.2: Results of 50 tests on the FRGC dataset with outlier removal

| Range Scan | | | Triangulation | | | Smoothing | | Decimation | | | | Cleanup | | | Registration | | | Deformation | |
|------------|--------|------------|---------------|--------|--------|-----------|------------------|------------|-------|-------|------------------|---------|-------|-------|--------------|------------------|---|-------------|------------------|
| ID | N_p | r_{\max} | t | N_v | N_t | t | e_{rms} | t | N_v | N_t | e_{rms} | t | N_v | N_t | t | e_{rms} | R | t | e_{rms} |
| 04774d74 | 79693 | 190.54 | 2.76 | 79693 | 156324 | 7.61 | 0.150 | 20.77 | 8267 | 15322 | 0.059 | 0.06 | 7614 | 13978 | 2.41 | 3.011 | Y | 4.38 | 1.011 |
| 04670d177 | 91612 | 175.70 | 2.65 | 91612 | 180080 | 8.35 | 0.151 | 25.93 | 9451 | 17691 | 0.050 | 0.06 | 6478 | 11964 | 1.85 | 3.000 | Y | 3.28 | 0.652 |
| 04343d331 | 91034 | 192.56 | 2.26 | 91034 | 178614 | 6.72 | 0.150 | 24.32 | 9471 | 17477 | 0.053 | 0.07 | 8581 | 15742 | 2.05 | 2.749 | Y | 2.94 | 0.501 |
| 04603d151 | 82747 | 164.20 | 2.39 | 82747 | 162396 | 7.43 | 0.139 | 22.52 | 8567 | 15933 | 0.047 | 0.06 | 7875 | 14544 | 2.01 | 2.864 | Y | 3.25 | 1.217 |
| 04530d319 | 110485 | 212.75 | 2.43 | 110485 | 215466 | 9.49 | 0.157 | 31.97 | 11601 | 20950 | 0.063 | 0.08 | 10666 | 18998 | 2.60 | 2.746 | Y | 3.16 | 0.533 |
| 04778d54 | 91996 | 239.06 | 3.03 | 91996 | 176382 | 8.00 | 0.175 | 23.87 | 9867 | 16962 | 0.084 | 0.06 | 5541 | 10317 | 1.78 | 2.433 | Y | 4.14 | 0.575 |
| 04461d293 | 106534 | 187.42 | 2.36 | 106534 | 209878 | 9.46 | 0.146 | 26.89 | 10946 | 20674 | 0.046 | 0.08 | 9198 | 17322 | 1.92 | 2.832 | Y | 3.01 | 1.259 |
| 04641d173 | 82338 | 176.91 | 2.75 | 82338 | 160166 | 7.31 | 0.163 | 20.36 | 8663 | 15575 | 0.066 | 0.06 | 7067 | 12969 | 2.04 | 2.850 | Y | 3.50 | 0.445 |
| 04667d194 | 72326 | 187.23 | 2.69 | 72326 | 141352 | 6.68 | 0.166 | 19.48 | 7581 | 13761 | 0.059 | 0.05 | 5255 | 9858 | 1.71 | 3.473 | Y | 3.31 | 0.925 |
| 04723d06 | 73378 | 212.14 | 3.17 | 73378 | 142954 | 6.78 | 0.198 | 20.02 | 7665 | 13956 | 0.084 | 0.05 | 6705 | 12086 | 2.08 | 3.467 | Y | 3.93 | 1.400 |
| 04831d50 | 91314 | 189.15 | 2.85 | 91314 | 179196 | 8.64 | 0.150 | 24.11 | 9497 | 17531 | 0.054 | 0.07 | 8247 | 15132 | 2.51 | 2.665 | Y | 3.95 | 0.543 |
| 04615d84 | 105448 | 222.59 | 2.72 | 105448 | 206880 | 9.51 | 0.160 | 29.56 | 10934 | 20270 | 0.055 | 0.09 | 10215 | 18607 | 2.68 | 3.283 | Y | 3.56 | 1.430 |
| 04771d06 | 112123 | 198.09 | 3.02 | 112123 | 220960 | 10.31 | 0.141 | 29.36 | 11583 | 21701 | 0.058 | 0.09 | 11038 | 20138 | 2.99 | 3.428 | Y | 4.62 | 0.683 |
| 04349d330 | 116305 | 212.14 | 2.31 | 116305 | 228890 | 8.68 | 0.135 | 32.16 | 11945 | 22540 | 0.047 | 0.09 | 10578 | 19714 | 2.19 | 3.244 | Y | 3.02 | 0.639 |
| 04684d230 | 113367 | 197.40 | 3.20 | 113367 | 222704 | 10.45 | 0.151 | 30.29 | 11748 | 21830 | 0.048 | 0.09 | 10417 | 19225 | 2.58 | 3.113 | Y | 3.64 | 0.951 |
| 04709d62 | 88770 | 180.24 | 2.74 | 88770 | 172890 | 7.85 | 0.156 | 24.84 | 9296 | 16863 | 0.060 | 0.06 | 6355 | 11754 | 1.91 | 3.542 | Y | 3.94 | 1.306 |
| 04777d88 | 104749 | 184.69 | 3.09 | 104749 | 203622 | 9.54 | 0.158 | 26.51 | 11013 | 19783 | 0.069 | 0.08 | 9888 | 16998 | 2.91 | 2.501 | Y | 4.47 | 0.591 |
| 04593d196 | 69103 | 150.87 | 2.33 | 69103 | 135890 | 6.26 | 0.154 | 17.49 | 7150 | 13337 | 0.051 | 0.05 | 6604 | 12144 | 1.78 | 3.225 | Y | 3.11 | 1.177 |
| 04596d90 | 76906 | 180.94 | 2.40 | 76906 | 151198 | 6.97 | 0.137 | 21.35 | 7931 | 14879 | 0.054 | 0.06 | 7257 | 13442 | 1.79 | 2.297 | Y | 3.22 | 0.438 |
| 04745d82 | 65374 | 178.90 | 2.99 | 65374 | 128064 | 6.07 | 0.147 | 16.72 | 6788 | 12546 | 0.052 | 0.04 | 5733 | 10593 | 1.69 | 1.968 | Y | 3.71 | 0.697 |
| 04588d129 | 94656 | 219.45 | 2.36 | 94656 | 185956 | 8.74 | 0.158 | 21.93 | 9803 | 18240 | 0.052 | 0.06 | 5951 | 11177 | 1.50 | 3.467 | Y | 3.01 | 1.328 |
| 02463d562 | 109120 | 190.76 | 2.35 | 109120 | 214936 | 8.18 | 0.153 | 24.50 | 11228 | 21146 | 0.048 | 0.10 | 9425 | 17629 | 1.79 | 2.714 | Y | 2.97 | 0.767 |
| 04222d391 | 107941 | 192.90 | 2.30 | 107941 | 212972 | 8.08 | 0.136 | 26.00 | 11088 | 20988 | 0.046 | 0.09 | 10616 | 19774 | 2.18 | 2.712 | Y | 2.98 | 0.621 |
| 04775d76 | 99636 | 197.48 | 2.99 | 99636 | 195416 | 9.75 | 0.142 | 26.54 | 10368 | 19105 | 0.046 | 0.07 | 6862 | 12762 | 1.86 | 2.618 | Y | 4.61 | 0.498 |
| 04297d261 | 92453 | 175.59 | 2.28 | 92453 | 180440 | 6.92 | 0.140 | 24.25 | 9689 | 17588 | 0.049 | 0.07 | 8762 | 16101 | 1.85 | 3.283 | Y | 2.88 | 1.359 |
| 04429d333 | 90939 | 162.55 | 2.30 | 90939 | 179432 | 7.82 | 0.137 | 20.91 | 9359 | 17663 | 0.043 | 0.07 | 8586 | 16004 | 2.03 | 3.713 | Y | 2.94 | 1.511 |
| 04661d167 | 93170 | 194.77 | 2.69 | 60941 | 118308 | 5.41 | 0.214 | 15.67 | 6449 | 11473 | 0.075 | 0.04 | 5176 | 9388 | 2.07 | 2.360 | Y | 3.24 | 0.696 |
| 04683d235 | 104426 | 209.15 | 2.79 | 104426 | 204946 | 9.30 | 0.141 | 24.49 | 10814 | 20110 | 0.045 | 0.08 | 8873 | 16656 | 2.29 | 3.345 | Y | 3.73 | 1.451 |
| 04385d335 | 77806 | 158.77 | 2.45 | 77806 | 152692 | 6.21 | 0.171 | 19.89 | 8053 | 14976 | 0.064 | 0.06 | 7411 | 13413 | 1.50 | 2.231 | Y | 2.86 | 0.755 |
| 04752d06 | 80842 | 169.46 | 3.09 | 80842 | 156666 | 7.08 | 0.142 | 20.49 | 8592 | 15169 | 0.062 | 0.06 | 6882 | 12810 | 1.80 | 2.608 | Y | 4.47 | 1.204 |
| 04801d64 | 65913 | 143.43 | 2.71 | 65913 | 129170 | 6.11 | 0.144 | 17.13 | 6869 | 12628 | 0.050 | 0.05 | 6015 | 11260 | 1.72 | 2.080 | Y | 4.28 | 0.423 |
| 04704d20 | 71871 | 159.89 | 2.86 | 71871 | 141096 | 6.65 | 0.150 | 19.54 | 7429 | 13858 | 0.059 | 0.05 | 6883 | 12581 | 2.05 | 2.573 | Y | 3.54 | 0.488 |

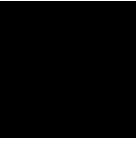
Table C.2: Results of 50 tests on the FRGC dataset with outlier removal

| Range Scan | | | Triangulation | | | Smoothing | | Decimation | | | | Cleanup | | | Registration | | | Deformation | |
|------------|--------|------------|---------------|--------|--------|-----------|------------------|------------|-------|-------|------------------|---------|-------|-------|--------------|------------------|---|-------------|------------------|
| ID | N_p | r_{\max} | t | N_v | N_t | t | e_{rms} | t | N_v | N_t | e_{rms} | t | N_v | N_t | t | e_{rms} | R | t | e_{rms} |
| 04237d155 | 74267 | 167.21 | 2.27 | 74267 | 145784 | 5.49 | 0.166 | 19.40 | 7682 | 14307 | 0.061 | 0.05 | 6756 | 12325 | 1.51 | 2.506 | Y | 2.81 | 0.478 |
| 04226d361 | 82287 | 163.70 | 2.29 | 82287 | 160816 | 6.05 | 0.143 | 20.87 | 8567 | 15733 | 0.050 | 0.06 | 7169 | 13316 | 1.58 | 3.264 | Y | 2.87 | 1.129 |
| 04632d144 | 100171 | 194.30 | 2.89 | 100171 | 197520 | 9.14 | 0.146 | 28.12 | 10331 | 19418 | 0.045 | 0.07 | 7362 | 13828 | 1.73 | 2.415 | Y | 3.36 | 0.732 |
| 04365d324 | 88440 | 190.21 | 2.28 | 88440 | 171564 | 6.53 | 0.147 | 24.14 | 9292 | 16666 | 0.055 | 0.06 | 7619 | 14239 | 1.57 | 3.240 | Y | 2.88 | 1.063 |
| 04372d271 | 66723 | 158.92 | 2.25 | 66723 | 129826 | 4.98 | 0.143 | 19.19 | 7003 | 12632 | 0.055 | 0.05 | 6334 | 11608 | 1.55 | 2.728 | Y | 2.85 | 1.072 |
| 04776d34 | 102197 | 185.59 | 2.90 | 102197 | 199480 | 9.49 | 0.156 | 26.60 | 10703 | 19423 | 0.061 | 0.07 | 7829 | 14409 | 1.95 | 2.605 | Y | 4.21 | 0.773 |
| 04830d96 | 106138 | 206.02 | 3.03 | 106138 | 207272 | 9.90 | 0.141 | 24.90 | 11081 | 20244 | 0.060 | 0.08 | 9942 | 17832 | 2.89 | 2.706 | Y | 4.07 | 0.485 |
| 04741d06 | 78825 | 155.80 | 2.76 | 78825 | 154200 | 7.22 | 0.143 | 22.19 | 8202 | 15094 | 0.056 | 0.06 | 7534 | 13773 | 2.07 | 3.367 | Y | 4.12 | 1.123 |
| 04388d295 | 102012 | 179.49 | 2.33 | 102012 | 201082 | 8.32 | 0.150 | 29.02 | 10496 | 19785 | 0.045 | 0.08 | 9908 | 18321 | 1.72 | 2.155 | Y | 2.96 | 0.691 |
| 04637d198 | 64187 | 166.08 | 2.73 | 64187 | 123676 | 6.04 | 0.155 | 14.35 | 6863 | 11931 | 0.060 | 0.05 | 5392 | 9800 | 1.76 | 3.310 | Y | 3.37 | 1.646 |
| 04743d56 | 116729 | 182.48 | 2.78 | 116729 | 228756 | 10.80 | 0.140 | 31.99 | 12144 | 22353 | 0.050 | 0.09 | 8690 | 16005 | 2.09 | 2.361 | Y | 4.20 | 0.797 |
| 04236d156 | 98398 | 193.06 | 2.30 | 98398 | 192476 | 7.26 | 0.146 | 25.78 | 10255 | 18794 | 0.063 | 0.07 | 8884 | 16227 | 2.01 | 3.454 | Y | 2.95 | 0.763 |
| 04822d48 | 83189 | 177.02 | 3.00 | 83189 | 163040 | 7.76 | 0.145 | 18.42 | 8655 | 15958 | 0.048 | 0.06 | 5715 | 10677 | 2.16 | 3.891 | Y | 3.86 | 1.002 |
| 04410d182 | 79164 | 165.32 | 2.31 | 93771 | 184304 | 8.09 | 0.141 | 27.12 | 9722 | 18076 | 0.046 | 0.07 | 8115 | 15346 | 1.72 | 2.622 | Y | 2.85 | 0.597 |
| 04797d52 | 98369 | 199.20 | 2.87 | 98369 | 192914 | 9.10 | 0.167 | 26.54 | 10216 | 18890 | 0.071 | 0.08 | 9236 | 16796 | 2.65 | 3.107 | Y | 4.01 | 0.991 |
| 04765d56 | 65713 | 139.98 | 2.89 | 65713 | 129148 | 6.03 | 0.150 | 16.92 | 6838 | 12628 | 0.045 | 0.05 | 6047 | 11349 | 1.97 | 2.893 | Y | 4.25 | 1.091 |
| 04387d330 | 100224 | 203.21 | 2.35 | 100224 | 196868 | 8.16 | 0.157 | 28.48 | 10367 | 19323 | 0.049 | 0.07 | 7117 | 13365 | 1.43 | 3.268 | Y | 2.81 | 1.181 |
| 04780d46 | 67854 | 166.30 | 2.84 | 67854 | 132794 | 6.07 | 0.151 | 17.68 | 7044 | 12999 | 0.056 | 0.05 | 5833 | 10677 | 1.80 | 2.230 | Y | 4.05 | 0.493 |

ID: Range scan file ID

 N_p : Number of points N_v : Number of vertices N_t : Number of triangles t : Computation time in seconds r_{\max} : Radius of bounding box, i.e., $r_{\max} = \sqrt{(x_{\max} - x_{\min})^2 + (y_{\max} - y_{\min})^2 + (z_{\max} - z_{\min})^2}/2$ e_{rms} : Root-mean-squared error, i.e., $e_{\text{rms}} = \sqrt{(d_1^2 + d_2^2 + \dots + d_n^2)/n}$

R: Results of registration (Y: successful; N: failed)



Qualitative Evaluation Questionnaire

This appendix presents the questionnaire for the qualitative evaluation of facial expression transferring, which is discussed in Section 5.3.

This page is intentionally left blank.

Qualitative Evaluation of Facial Expression Transferring

1. Introduction

Thank you for participating in the qualitative evaluation of facial expression transferring. This evaluation will take about 20 minutes to complete.

In this evaluation, you are about to see 15 groups of faces with 6 expressions for each group. Your task is to rate how similar in facial expression it is for each pair of faces.













2. User Information

Name:

Gender: ☐ Male ☐ Female













Group 1

How would you rate the similarity of facial expressions in the following faces?

| Expression | Template | Boy | Similarity of Expression |
|------------|---|---|--|
| Anger |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Laughing |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Pleased |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Rage |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Sad |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Surprise |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |













Group 2

How would you rate the similarity of facial expressions in the following faces?

| Expression | Template | Girl | Similarity of Expression |
|------------|---|---|--|
| Anger |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Laughing |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Pleased |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Rage |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Sad |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Surprise |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |













Group 3

How would you rate the similarity of facial expressions in the following faces?

| Expression | Template | Judy | Similarity of Expression |
|------------|---|---|--|
| Anger |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Laughing |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Pleased |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Rage |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Sad |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Surprise |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |













Group 4

How would you rate the similarity of facial expressions in the following faces?

| Expression | Template | Penny | Similarity of Expression |
|------------|---|---|--|
| Anger |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Laughing |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Pleased |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Rage |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Sad |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Surprise |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |













Group 5

How would you rate the similarity of facial expressions in the following faces?

| Expression | Template | Will | Similarity of Expression |
|------------|---|---|--|
| Anger |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Laughing |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Pleased |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Rage |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Sad |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Surprise |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |













Group 6

How would you rate the similarity of facial expressions in the following faces?

| Expression | Template | Woman | Similarity of Expression |
|------------|---|---|--|
| Anger |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Laughing |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Pleased |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Rage |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Sad |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Surprise |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |













Group 7

How would you rate the similarity of facial expressions in the following faces?

| Expression | Template | 02463d550 | Similarity of Expression |
|------------|---|---|--|
| Anger |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Laughing |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Pleased |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Rage |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Sad |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Surprise |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |













Group 8

How would you rate the similarity of facial expressions in the following faces?

| Expression | Template | 04201d368 | Similarity of Expression |
|------------|---|---|--|
| Anger |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Laughing |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Pleased |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Rage |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Sad |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Surprise |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |













Group 9

How would you rate the similarity of facial expressions in the following faces?

| Expression | Template | 04213d280 | Similarity of Expression |
|------------|---|---|--|
| Anger |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Laughing |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Pleased |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Rage |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Sad |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Surprise |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |













Group 10

How would you rate the similarity of facial expressions in the following faces?

| Expression | Template | 04225d299 | Similarity of Expression |
|------------|---|---|--|
| Anger |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Laughing |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Pleased |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Rage |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Sad |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Surprise |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |













Group 11

How would you rate the similarity of facial expressions in the following faces?

| Expression | Template | 04300d222 | Similarity of Expression |
|------------|---|---|--|
| Anger |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Laughing |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Pleased |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Rage |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Sad |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Surprise |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |













Group 12

How would you rate the similarity of facial expressions in the following faces?

| Expression | Template | 04336d297 | Similarity of Expression |
|------------|---|---|--|
| Anger |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Laughing |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Pleased |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Rage |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Sad |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Surprise |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |













Group 13

How would you rate the similarity of facial expressions in the following faces?

| Expression | Template | 04343d323 | Similarity of Expression |
|------------|---|---|--|
| Anger |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Laughing |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Pleased |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Rage |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Sad |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Surprise |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |













Group 14

How would you rate the similarity of facial expressions in the following faces?

| Expression | Template | 04410d182 | Similarity of Expression |
|------------|---|---|--|
| Anger |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Laughing |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Pleased |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Rage |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Sad |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Surprise |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |

Group 15

How would you rate the similarity of facial expressions in the following faces?

| Expression | Template | 04485d284 | Similarity of Expression |
|------------|---|---|--|
| Anger |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Laughing |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Pleased |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Rage |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Sad |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |
| Surprise |  |  | <input type="checkbox"/> Very Dissimilar <input type="checkbox"/> Dissimilar <input type="checkbox"/> Neutral <input type="checkbox"/> Similar <input type="checkbox"/> Very Similar |

Bibliography

- ABBOUD, B., DAVOINE, F., AND DANG, M. Statistical modeling for facial expression analysis and synthesis. In *ICIP '03: Proceedings of the 2003 IEEE International Conference on Image Processing*, volume 1, pages I-653-6, Barcelona, Catalonia, Spain, 14-18 Sept. 2003. IEEE Computer Society. doi: 10.1109/ICIP.2003.1247046. (Cited on page 8.)
- ABRANTES, G. A. AND PEREIRA, F. MPEG-4 facial animation technology: Survey, implementation, and results. *IEEE Transactions on Circuits and Systems for Video Technology*, 9(2):290-305, Mar. 1999. doi: 10.1109/76.752096. (Cited on page 7.)
- ALBRECHT, I., HABER, J., KÄHLER, K., SCHRÖDER, M., AND SEIDEL, H.-P. “may i talk to you? :-)” — facial animation from text. In *PG '02: Proceedings of the 10th Pacific Conference on Computer Graphics and Applications*, pages 77-86, Beijing, China, 9-11 Oct. 2002. IEEE Computer Society. doi: 10.1109/PCCGA.2002.1167841. (Cited on page 8.)
- ALEXA, M., COHEN-OR, D., AND LEVIN, D. As-rigid-as-possible shape interpolation. In *SIGGRAPH '00: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, pages 157-164, New Orleans, Louisiana, USA, 23-28 July 2000. ACM. doi: 10.1145/344779.344859. (Cited on page 18.)
- ALEXANDER, O., ROGERS, M., LAMBETH, W., CHIANG, M., AND DEBEVEC, P. The digital emily project: photoreal facial modeling and animation. In *SIGGRAPH '09: ACM SIGGRAPH 2009 Courses*, pages 12:1-15, New Orleans, Louisiana, USA, 3-7 Aug. 2009. ACM. doi: 10.1145/1667239.1667251. (Cited on page 2.)
- ALLEN, B., CURLESS, B., AND POPOVIĆ, Z. The space of human body shapes: reconstruction and parameterization from range scans. In *SIGGRAPH '03: ACM SIGGRAPH 2003 Papers*, pages 587-594, San Diego, California, USA, 27-31 July 2003. ACM. doi: 10.1145/1201775.882311. (Cited on pages 2, 14, 16, and 22.)
- BÆRENTZEN, J. A. AND AANÆS, H. Signed distance computation using the angle weighted pseudonormal. *IEEE Transactions on Visualization and Computer Graphics*, 11(3):243-253, May 2005. doi: 10.1109/TVCG.2005.49. (Cited on page 69.)
- BALLARD, D. H. Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111-122, 1981. doi: 10.1016/0031-3203(81)90009-1. (Cited on page 30.)

- BAR-ITZHACK, I. Y. New method for extracting the quaternion from a rotation matrix. *AIAA Journal of Guidance, Control and Dynamics*, 23(6):1085–1087, 2000. doi: 10.2514/2.4654. (Cited on page 64.)
- BASCLE, B. AND BLAKE, A. Separability of pose and expression in facial tracking and animation. In *ECCV '98: Proceedings of the 5th European Conference on Computer Vision*, pages 323–328, Freiburg, Germany, 2–6 June 1998. Springer. doi: 10.1109/ICCV.1998.710738. (Cited on page 27.)
- BEELER, T., HAHN, F., BRADLEY, D., BICKEL, B., GOTSMAN, C., SUMNER, R. W., AND GROSS, M. High-quality passive facial performance capture using anchor frames. In *SIGGRAPH '11: ACM SIGGRAPH 2011 Papers*, pages 75:1–75:10, Vancouver, British Columbia, Canada, 7–11 Aug. 2011. ACM. doi: 10.1145/1964921.1964970. (Cited on pages 2, 8, 10, 13, 14, and 18.)
- BELONGIE, S., MALIK, J., AND PUZICHA, J. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, Apr. 2002. doi: 10.1109/34.993558. (Cited on page 21.)
- BEN-CHEN, M., WEBER, O., AND GOTSMAN, C. Spatial deformation transfer. In *SCA '09: Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 67–74, New Orleans, Louisiana, USA, 1–2 Aug. 2009. ACM. doi: 10.1145/1599470.1599479. (Cited on page 2.)
- BENTLEY, J. L. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9):509–517, Sept. 1975. doi: 10.1145/361002.361007. (Cited on page 34.)
- BERNARDINI, F. AND RUSHMEIER, H. The 3D model acquisition pipeline. *Computer Graphics Forum*, 21(2):149–172, June 2002. doi: 10.1111/1467-8659.00574. (Cited on pages 2, 14, 18, and 19.)
- BESL, P. J. AND MCKAY, N. D. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, Feb. 1992. doi: 10.1109/34.121791. (Cited on pages 19, 30, and 72.)
- BLANZ, V. AND VETTER, T. A morphable model for the synthesis of 3D faces. In *SIGGRAPH '99: Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, pages 187–194, Los Angeles, California, USA, 8–13 Aug. 1999. ACM. doi: 10.1145/311535.311556. (Cited on pages 1, 7, 9, 14, 15, and 16.)
- BLANZ, V., BASSO, C., POGGIO, T., AND VETTER, T. Reanimating faces in images and video. In *EUROGRAPHICS '03: Proceedings of the 24th Annual Conference of the Eurographics Association*, pages 641–650, Granada, Spain, 1–6 Sept. 2003. Eurographics Association. doi: 10.1111/1467-8659.t01-1-00712. (Cited on pages 1, 7, 9, 14, and 16.)
- BOOKSTEIN, F. L. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6):567–585, June 1989. doi: 10.1109/34.24792. (Cited on pages 22 and 23.)

- BOTSCH, M. AND KOBELT, L. An intuitive framework for real-time freeform modeling. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, pages 630–634, Los Angeles, California, USA, 8–12 Aug. 2004. ACM. doi: 10.1145/1186562.1015772. (Cited on page 25.)
- BOTSCH, M. AND PAULY, M. Geometric modeling based on polygonal meshes. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Courses*, page 29, San Diego, California, USA, 30 July–3 Aug. 2006. ACM. (Cited on pages 25 and 32.)
- BOTSCH, M. AND SORKINE, O. On linear variational surface deformation methods. *IEEE Transactions on Visualization and Computer Graphics*, 14(1):213–230, Jan./Feb. 2008. doi: 10.1109/TVCG.2007.1054. (Cited on pages 2, 25, and 37.)
- BOTSCH, M., SUMNER, R. W., PAULY, M., AND GROSS, M. Deformation transfer for detail-preserving surface editing. In *VMV '06: Proceedings of the Vision, Modeling and Visualization Conference 2006*, pages 257–364, Aachen, Germany, 22–24 Nov. 2006. Aka GmbH. (Cited on pages 25 and 26.)
- BOTSCH, M., PAULY, M., KOBELT, L., ALLIEZ, P., LÉVY, B., BISCHOFF, S., AND RÖSSL, C. Geometric modeling based on polygonal meshes. In *SIGGRAPH '07: ACM SIGGRAPH 2007 Courses*, pages 23:1–181, San Diego, California, USA, 5–9 Aug. 2007. ACM. url: <http://doi.acm.org/10.1145/1281500.1281640>. (Cited on pages 18, 25, and 83.)
- BOUREL, F., CHIBELUSHI, C. C., AND LOW, A. A. Robust facial feature tracking. In *BMVC '00: Proceedings of 11th British Machine Vision Conference*, pages 232–241, 2000. (Cited on page 27.)
- BROWN, B. J. AND RUSINKIEWICZ, S. Global non-rigid alignment of 3-D scans. In *SIGGRAPH '07: ACM SIGGRAPH 2007 Papers*, pages 21:1–9, San Diego, California, USA, 5–9 Aug. 2007. ACM. doi: 10.1145/1276377.1276404. (Cited on page 22.)
- CARR, J. C., BEATSON, R. K., CHERRIE, J. B., MITCHELL, T. J., FRIGHT, W. R., MCCALLUM, B. C., AND EVANS, T. R. Reconstruction and representation of 3d objects with radial basis functions. In *SIGGRAPH 01': Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, pages 67–76, Los Angeles, California, USA, 12–17 Aug. 2001. ACM. doi: 10.1145/383259.383266. (Cited on page 23.)
- CGAL. Computational geometry algorithms library. url: <http://www.cgal.org/>. (Cited on page 33.)
- CHELLAPPA, R., DU, M., TURAGA, P., AND ZHOU, S. K. Face tracking and recognition in video. In LI, S. Z. AND JAIN, A. K., editors, *Handbook of Face Recognition*, pages 323–351. Springer, 2011. doi: 10.1007/978-0-85729-932-1_13. (Cited on page 6.)
- CHEN, C.-S., HUNG, Y.-P., AND CHENG, J.-B. A fast automatic method for registration of partially-overlapping range images. In *ICCV '98: Proceedings of the 6th IEEE International Conference on Computer Vision*, pages 242–248, Bombay, India, 4–7 Jan. 1998. IEEE Computer Society. doi: 10.1109/ICCV.1998.710725. (Cited on page 20.)

- CHEN, Y. AND MEDIONI, G. Object modelling by registration of multiple range images. *Image and Vision Computing*, 10(3):145–155, Apr. 1992. doi: 10.1016/0262-8856(92)90066-C. (Cited on page 72.)
- CHOW, C. K., TSUI, H. T., AND LEE, T. Surface registration using a dynamic genetic algorithm. *Pattern Recognition*, 37(1):105–117, Jan. 2004. doi: 10.1016/S0031-3203(03)00222-X. (Cited on page 72.)
- CHUI, H. AND RANGARAJAN, A. A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding*, (2–3):114–141, Feb. 2003. doi: 10.1016/S1077-3142(03)00009-2. (Cited on pages 21 and 23.)
- COOTES, T. F., EDWARDS, G. J., AND TAYLOR, C. J. Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):681–685, June 2001. doi: 10.1109/34.927467. (Cited on page 27.)
- COUDRIN, B., DEVY, M., ORTEU, J.-J., AND BRÈTHES, L. An innovative hand-held vision-based digitizing system for 3d modelling. *Optics and Lasers in Engineering*, 49(9–10):1168–1176, Sept. 2011. doi: 10.1016/j.optlaseng.2011.05.004. (Cited on pages 2, 10, and 14.)
- CRISTINACCE, D. AND COOTES, T. F. Facial feature detection and tracking with automatic template selection. In *FGR '06: Proceedings of the 7th IEEE International Conference on Automatic Face and Gesture Recognition*, pages 429–434, Southampton, UK, 10–12 Apr. 2006. IEEE Computer Society. doi: 10.1109/FGR.2006.50. (Cited on page 27.)
- CURLESS, B. From range scans to 3D models. *ACM SIGGRAPH Computer Graphics*, 33(4):38–41, Nov. 2000. doi: 10.1145/345370.345399. (Cited on pages 2 and 14.)
- DARWIN, C. *The Expression of the Emotions in Man and Animals*. John Murray, London, 1872. (Cited on page 5.)
- DECARLO, D. AND METAXAS, D. Optical flow constraints on deformable models with applications to face tracking. *International Journal of Computer Vision*, 38(2):99–127, July 2000. doi: 10.1023/A:1008122917811. (Cited on pages 3, 13, 27, and 81.)
- DECARLO, D., METAXAS, D., AND STONE, M. An anthropometric face model using variational techniques. In *SIGGRAPH '98: Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, pages 67–74, Orlando, Florida, USA, 19–24 July 1998. ACM. doi: 10.1145/280814.280823. (Cited on page 7.)
- DENG, Z. AND NEUMANN, U. eFASE: Expressive facial animation synthesis and editing with phoneme-isomap controls. In *SCA '06: Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 251–260, Vienna, Austria, 2–4 Sept. 2006. ACM. doi: 10.2312/SCA/SCA06/251-259. (Cited on page 8.)
- DORAI, C., WENG, J., AND JAIN, A. K. Optimal registration of object views using range data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(10):1131–1138, Oct. 1997. doi: 10.1109/34.625115. (Cited on page 20.)

- EKMAN, P. AND FRIESEN, W. *Facial action coding system: A technique for the measurement of facial movement*. Consulting Psychologists Press, 1978. (Cited on pages 6 and 11.)
- EKMAN, P. AND FRIESEN, W. V. Constants across cultures in the face and emotion. *Journal of Personality and Social Psychology*, 17(2):124–129, Feb. 1971. doi: 10.1037/h0030377. (Cited on page 5.)
- ERSOTELOS, N. AND DONG, F. Building highly realistic facial modeling and animation: A survey. *The Visual Computer*, 24(1):13–30, Jan. 2008. doi: 10.1007/s00371-007-0175-y. (Cited on pages 8 and 9.)
- FANG, T., ZHAO, X., OCEGUEDA, O., SHAH, S. K., AND KAKADIARIS, I. A. 3D/4D facial expression analysis: An advanced annotated face model approach. *Image and Vision Computing*, 30(10):738–749, Oct. 2012. doi: 10.1016/j.imavis.2012.02.004. (Cited on page 6.)
- FASEL, B. AND LUETTIN, J. Automatic facial expression analysis: A survey. *Pattern Recognition*, 36(1):259–275, Jan. 2003. doi: 10.1016/S0031-3203(02)00052-3. (Cited on pages 6 and 27.)
- FRÖHLICH, S. AND BOTSCH, M. Example-driven deformation based on discrete shells. *Computer Graphics Forum*, 30(8):2246–2257, Dec. 2011. doi: 10.1111/j.1467-8659.2011.01974.x. (Cited on page 25.)
- GAL, R. AND COHEN-OR, D. Salient geometric features for partial shape matching and similarity. *Transactions of Graphics*, 25(1):130–150, Jan. 2006. doi: 10.1145/1122501.1122507. (Cited on page 81.)
- GARLAND, M. AND HECKBERT, P. S. Surface simplification using quadric error metrics. In *SIGGRAPH '97: Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, pages 209–216, Los Angeles, California, USA, 3–8 Aug. 1997. ACM. doi: 10.1145/258734.258849. (Cited on page 33.)
- GELFAND, N., RUSINKIEWICZ, S., IKEMOTO, L., AND LEVOY, M. Geometrically stable sampling for the ICP algorithm. In *3DIM '03: Proceedings of the 4th International Conference on 3D Digital Imaging and Modeling*, pages 260–267, Banff, Canada, 6–10 Oct. 2003. IEEE Computer Society. doi: 10.1109/IM.2003.1240258. (Cited on page 20.)
- GELFAND, N., MITRA, N. J., GUIBAS, L. J., AND POTTMANN, H. Robust global registration. In *SGP '05: Proceedings of the 3rd Eurographics Symposium on Geometry Processing*, pages 197–206, Vienna, Austria, 4–6 July 2005. Eurographics Association. (Cited on pages 20, 21, and 30.)
- GIBSON, S. F. F. AND MIRTICH, B. A survey of deformable modeling in computer graphics. Technical Report TR-97-19, MERL, Nov. 1997. (Cited on page 24.)
- GOLD, S., RANGARAJAN, A., LU, C.-P., PAPPU, S., AND MJOLSNESS, E. New algorithms for 2d and 3d point matching: pose estimation and correspondence. *Pattern Recognition*, 31(8):1019–1031, Aug. 1998. doi: 10.1016/S0031-3203(98)80010-1. (Cited on pages 21 and 23.)

- GRIESSER, R. T., CUNNINGHAM, D. W., WALLRAVEN, C., AND BÜLTHOFF, H. H. Psychophysical investigation of facial expressions using computer animated faces. In WALLRAVEN, C. AND SUNDSTEDT, V., editors, *APGV '07: Proceedings of the 4th Symposium on Applied Perception in Graphics and Visualization*, pages 11–18, Tübingen, Germany, 25–27 July 2007. ACM. doi: 10.1145/1272582.1272585. (Cited on page 1.)
- GUENTER, B., GRIMM, C., WOOD, D., MALVAR, H., AND PIGHIN, F. Making faces. In *SIGGRAPH '98: Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, pages 55–66, Orlando, Florida, USA, 19–24 July 1998. ACM. doi: 10.1145/280814.280822. (Cited on pages 7, 12, and 14.)
- HÄHNEL, D., THRUN, S., AND BURGARD, W. An extension of the ICP algorithm for modeling nonrigid objects with mobile robots. In *IJCAI '03: Proceedings of the 18th International Joint Conference on Artificial Intelligence*, pages 915–920, Acapulco, Mexico, 9–15 Aug. 2003. Springer. (Cited on page 22.)
- HJELMÅS, E. AND LOW, B. K. Face detection: A survey. *Computer Vision and Image Understanding*, 83(3):236–274, Sept. 2001. doi: 10.1006/cviu.2001.0921. (Cited on page 6.)
- HO, J., YANG, M.-H., RANGARAJAN, A., AND VEMURI, B. A new affine registration algorithm for matching 2D point sets. In *WACV '07: IEEE Workshop on Applications of Computer Vision*, Austin, TX, Feb. 2007. doi: 10.1109/WACV.2007.6. (Cited on page 21.)
- HOPPE, H., DEROSE, T., DUCHAMP, T., McDONALD, J., AND STUETZLE, W. Surface reconstruction from unorganized points. In *SIGGRAPH '92: Proceedings of the 19th Annual Conference on Computer Graphics and Interactive Techniques*, pages 71–78, Chicago, Illinois, USA, 26–31 July 1992. ACM. doi: 10.1145/133994.134011. (Cited on page 32.)
- HOPPE, H., DEROSE, T., DUCHAMP, T., McDONALD, J., AND STUETZLE, W. Mesh optimization. In *SIGGRAPH '93: Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques*, pages 19–26, Anaheim, California, USA, 2–6 Aug. 1993. ACM. doi: 10.1145/166117.166119. (Cited on page 83.)
- HORMANN, K., POLTHIER, K., AND SHEFFER, A. Mesh parameterization: Theory and practice. In *SIGGRAPH Asia '08: ACM SIGGRAPH Asia 2008 Courses*, pages 47:1–87, Singapore, 10–13 Dec. 2008. ACM. doi: 10.1145/1508044.1508091. (Cited on page 83.)
- HORN, B. K. P. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A*, 4(4):629–642, Apr. 1987. doi: 10.1364/JOSAA.4.000629. (Cited on page 34.)
- HORN, B. K. P., HILDEN, H. M., AND NEGAHDARIPOUR, S. Closed-form solution of absolute orientation using orthonormal matrices. *Journal of the Optical Society of America A*, 5(7):1127–1135, July 1988. doi: 10.1364/JOSAA.5.001127. (Cited on page 34.)

- HUANG, H., CHAI, J., TONG, X., AND WU, H.-T. Leveraging motion capture and 3D scanning for high-fidelity facial performance acquisition. In *SIGGRAPH '11: ACM SIGGRAPH 2011 Papers*, pages 74:1–74:10, Vancouver, British Columbia, Canada, 7–11 Aug. 2011. ACM. doi: 10.1145/2010324.1964969. (Cited on page 13.)
- JOHNSON, A. E. AND HEBERT, M. Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):433–449, May 1999. doi: 10.1109/34.765655. (Cited on pages 20 and 72.)
- JU, X. AND SIEBERT, J. P. Conforming generic animatable models to 3d scanned data. In *International Conference of Numberisation 3D-Scanning*, 2001. (Cited on page 23.)
- KÄHLER, K., HABER, J., AND SEIDEL, H.-P. Geometry-based muscle modeling for facial animation. In *GI '01: Proceedings of the Graphics Interface 2001 Conference*, pages 37–46, Ottawa, Ontario, Canada, 7–9 June 2001. Canadian Human-Computer Communications Society. (Cited on page 6.)
- KALBERER, G. A. AND GOOL, L. V. Face animation based on observed 3D speech dynamics. In *CA '01: Proceedings of Computer Animation 2001*, pages 20–27, Seoul, Korea, 7–8 Nov. 2001. IEEE Computer Society. doi: 10.1109/CA.2001.982373. (Cited on page 8.)
- KASS, M., WITKIN, A., AND TERZOPOULOS, D. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, Jan. 1988. doi: 10.1007/BF00133570. (Cited on pages 24 and 27.)
- KRAEVOY, V. AND SHEFFER, A. Cross-parameterization and compatible remeshing of 3D models. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, pages 861–869, Los Angeles, California, USA, 8–12 Aug. 2004. ACM. doi: 10.1145/1186562.1015811. (Cited on page 18.)
- LAMDAN, Y. AND WOLFSON, H. J. Geometric hashing: A general and efficient model-based recognition scheme. In *ICCV '88: Proceedings of the 2nd IEEE International Conference on Computer Vision*, pages 238–249, Florida, USA, 5–8 Dec. 1988. IEEE Computer Society. doi: 10.1109/CCV.1988.589995. (Cited on page 30.)
- LANMAN, D. AND TAUBIN, G. Build your own 3D scanner: 3D photography for beginners. In *SIGGRAPH '09: ACM SIGGRAPH 2009 Courses*, pages 8:1–94, New Orleans, Louisiana, USA, 3–7 Aug. 2009. ACM. doi: 10.1145/1667239.1667247. (Cited on pages 2, 10, and 14.)
- LAVAGETTO, F. AND POCKAJ, R. The facial animation engine: toward a high-level interface for the design of MPEG-4 compliant. *IEEE Transactions on Circuits and Systems for Video Technology*, 9(2):277–289, Mar. 1999. doi: 10.1109/76.752095. (Cited on page 7.)
- LEE, Y., TERZOPOULOS, D., AND WATERS, K. Constructing physics-based facial models of individuals. In *GI '93: Proceedings of the Graphics Interface 1993 Conference*, pages 1–8, Toronto, Ontario, Canada, 19–21 May 1993. Canadian Human-Computer Communications Society. (Cited on pages 1, 6, 9, 10, and 11.)

- LEE, Y., TERZOPOULOS, D., AND WATERS, K. Realistic modeling for facial animation. In *SIGGRAPH '95: Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, pages 55–62, Los Angeles, California, USA, 6–11 Aug. 1995. ACM. doi: 10.1145/218380.218407. (Cited on pages 1, 6, 9, 10, and 11.)
- LEVOY, M., PULLI, K., CURLESS, B., RUSINKIEWICZ, S., KOLLER, D., PEREIRA, L., GINTON, M., ANDERSON, S., DAVIS, J., GINSBERG, J., SHADE, J., AND FULK, D. The digital Michelangelo project: 3D scanning of large statues. In *SIGGRAPH '00: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, pages 131–144, New Orleans, Louisiana, USA, 23–28 July 2000. ACM. doi: 10.1145/344779.344849. (Cited on page 20.)
- LEWIS, J. P. AND ANJYO, K. Direction-manipulation blendshapes. *IEEE Computer Graphics and Applications*, 30(4):42–50, July 2010. doi: 10.1109/MCG.2010.41. (Cited on page 10.)
- LI, H., ADAMS, B., GUIBAS, L. J., AND PAULY, M. Robust single-view geometry and motion reconstruction. In *SIGGRAPH Asia '09: ACM SIGGRAPH Asia 2009 Papers*, pages 175:1–175:10, Yokohama, Japan, 16–19 Dec. 2009. ACM. doi: 10.1145/1618452.1618521. (Cited on pages 2 and 23.)
- LI, H., WEISE, T., AND PAULY, M. Example-based facial rigging. In *SIGGRAPH '10: ACM SIGGRAPH 2010 Papers*, pages 32:1–32:6, Los Angeles, CA, USA, 26–30 July 2010. ACM. doi: 10.1145/1833349.1778769. (Cited on pages 1, 8, 9, 14, and 16.)
- LI, H., LUO, L., VLASIC, D., PEERS, P., POPOVIĆ, J., PAULY, M., AND RUSINKIEWICZ, S. Temporally coherent completion of dynamic shapes. *ACM Transactions on Graphics*, 31(1):2:1–2:11, Feb. 2012. doi: 10.1145/2077341.2077343. (Cited on page 23.)
- LINDSTROM, P. AND TURK, G. Fast and memory efficient polygonal simplification. In *Vis '98: Proceedings of the 9th IEEE Visualization Conference*, pages 279–286, Research Triangle Park, NC, USA, 18–23 Oct. 1998. IEEE Computer Society. doi: 10.1109/VISUAL.1998.745314. (Cited on page 33.)
- LIPMAN, Y., SORKINE, O., COHEN-OR, D., LEVIN, D., RÖSSL, C., AND SEIDEL, H.-P. Differential coordinate for interactive mesh editing. In *SMI '04: Proceedings of the 2004 IEEE International Conference on Shape Modeling and Applications*, pages 181–190, Genova, Italy, 7–9 June 2004. IEEE Computer Society. doi: 10.1109/SMI.2004.1314505. (Cited on page 25.)
- LONCARIC, S. A survey of shape analysis techniques. *Pattern Recognition*, 31(8): 983–1001, Aug. 1998. doi: 10.1016/S0031-2023(97)00122-2. (Cited on page 18.)
- LORENSEN, W. E. AND CLINE, H. E. Marching cubes: A high resolution 3D surface construction algorithm. In *SIGGRAPH '87: Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, pages 163–169, Anaheim, California, USA, 27–31 July 1987. ACM. doi: 10.1145/37401.37422. (Cited on page 32.)
- LOWE, D. G. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, Nov. 2004. doi: 10.1023/B:VISI.0000029664.99615.94. (Cited on page 81.)

- LUO, B. AND HANCOCK, E. R. Structural graph matching using the em algorithm and singular value decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10):1120–1136, Oct. 2001. doi: 10.1109/34.954602. (Cited on page 21.)
- MAGNENAT-THALMANN, N., PRIMEAU, E., AND THALMANN, D. Abstract muscle action procedures for human face animation. *The Visual Computer*, 3(5):290–297, Mar. 1988. doi: 10.1007/BF01914864. (Cited on pages 6, 11, and 12.)
- MAINTZ, J. B. A. AND VIERGEVER, M. A. A survey of medical image registration. *Medical Image Analysis*, 2(1):1–36, Mar. 1998. doi: 10.1016/S1361-8415(01)80026-8. (Cited on page 18.)
- MCINERNEY, T. AND TERZOPOULOS, D. Deformable models in medical image analysis: A survey. *Medical Image Analysis*, 1(2):91–108, June 1996. doi: 10.1016/S1361-8415(96)80007-7. (Cited on page 24.)
- MEYER, M., DESBRUN, M., SCHRÖDER, P., AND BARR, A. H. Discrete differential geometry operators for triangulated 2-manifolds. In *VisMath '02*, 2002. url: <http://www.cs.caltech.edu/~mmeyer/Publications/diffGeomOps.pdf>. (Cited on pages 91, 93, and 94.)
- MOON, T. K. The expectation-maximization algorithm. *IEEE Signal Processing Magazine*, 13(6):47–60, Nov. 1996. doi: 10.1109/79.543975. (Cited on page 21.)
- MYRONENKO, A. AND SONG, X. Point set registration: Coherent point drift. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(12):2262–2275, Dec. 2010. doi: 10.1109/TPAMI.2010.46. (Cited on page 21.)
- NEALEN, A., MÜLLER, M., KEISER, R., BOXERMAN, E., AND CARLSON, M. Physically based deformable models in computer graphics. *Computer Graphics Forum*, 25(4):809–836, Dec. 2006. doi: 10.1111/j.1467-8659.2006.01000.x. (Cited on pages 24 and 25.)
- NOH, J.-Y. AND NEUMANN, U. Expression cloning. In *SIGGRAPH 01': Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, pages 277–288, Los Angeles, California, USA, 12–17 Aug. 2001. ACM. doi: 10.1145/383259.383290. (Cited on pages 2, 8, 26, and 54.)
- ORVALHO, V., BASTOS, P., PARKE, F., OLIVEIRA, B., AND ALVAREZ, X. A facial rigging survey. In *EG 2012 – State of the Art Reports*, pages 183–204, Cagliari, Sardinia, Italy, 2012. Eurographics Association. doi: 10.2312/conf/EG2012/stars/183-204. (Cited on pages 1 and 7.)
- OSHUA PODOLAK, SHILANE, P., GOLOVINSKIY, A., RUSINKIEWICZ, S., AND FUNKHOUSER, T. A planar-reflective symmetry transform for 3D shapes. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Papers*, pages 549–559, Boston, MA, USA, July 2006. ACM. doi: 10.1145/1179352.1141923. (Cited on page 20.)
- OSTERMANN, J. Animation of synthetic faces in MPEG-4. In *CA '98: Proceedings of Computer Animation 1998*, pages 49–55, Philadelphia, PA, USA, 8–10 June 1998. IEEE Computer Society. doi: 10.1109/CA.1998.681907. (Cited on page 7.)

- PARKE, F. I. Computer generated animation of faces. In *ACM '72: Proceedings of the ACM Annual Conference*, pages 451–457, Boston, Massachusetts, USA, 1 Aug. 1972. ACM. doi: 10.1145/800193.569955. (Cited on pages 1 and 9.)
- PARKE, F. I. *A parametric model for human faces*. PhD thesis, University of Utah, Dec. 1974. (Cited on pages 1, 6, and 9.)
- PARKE, F. I. Parameterized models for facial animation. *IEEE Computer Graphics and Applications*, 2:61–68, Nov. 1982. doi: 10.1109/MCG.1982.1674492. (Cited on pages 1, 9, 10, and 11.)
- PARKE, F. I. AND WATERS, K. *Computer Facial Animation*. A K Peters Ltd, 2nd edition, 13 Nov. 2008. ISBN: 1568814488. (Cited on pages 1, 6, and 9.)
- PEKELNY, Y. AND GOTSMAN, C. Articulated object reconstruction and markerless motion capture from depth video. *Computer Graphics Forum*, 27(2):299–408, Apr. 2008. doi: 10.1111/j.1467-8659.2008.01137.x. (Cited on pages 23 and 28.)
- PETERSEN, K. B. AND PEDERSEN, M. S. The matrix cookbook, 2008. url: <http://matrixcookbook.com>. (Cited on page 59.)
- PHILLIPS, P. J., FLYNN, P. J., SCRUGGS, W. T., BOWYER, K. W., CHANG, J., HOFFMAN, K. J., MARQUES, J., MIN, J., AND WOREK, W. J. Overview of the face recognition grand challenge. In *CVPR '05: Proceedings of the 2005 IEEE Conference on Computer Vision and Pattern Recognition*, pages 947–954, San Diego, CA, USA, 20–26 June 2005. IEEE Computer Society. doi: 10.1109/CVPR.2005.268. (Cited on page 31.)
- PIGHIN, F. Performance-driven facial animation. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Courses*, page 30, San Diego, California, USA, 30 July–3 Aug. 2006. ACM. (Cited on page 13.)
- PIGHIN, F., HECKER, J., LISCHINSKI, D., SZELISKI, R., AND SALESIN, D. H. Synthesizing realistic facial expressions from photographs. In *SIGGRAPH '98: Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, pages 75–84, Orlando, Florida, USA, 19–24 July 1998. ACM. doi: 10.1145/280814.280825. (Cited on pages 1, 7, 9, and 12.)
- PLATT, S. M. AND BADLER, N. I. Animating facial expressions. In *SIGGRAPH '81: Proceedings of the 8th Annual Conference on Computer Graphics and Interactive Techniques*, pages 245–252, Dallas, Texas, USA, 3–7 Aug. 1981. ACM. doi: 10.1145/800224.806812. (Cited on pages 1, 6, 9, 10, and 11.)
- PULLI, K. Multiview registration for large data sets. In *3DIM '99: Proceedings of the 2nd International Conference on 3D Digital Imaging and Modeling*, pages 160–168, Ottawa, Canada, 4–8 Oct. 1999. IEEE Computer Society. doi: 10.1109/IM.1999.805346. (Cited on pages 20 and 72.)
- PYUN, H., KIM, Y., CHAE, W., KANG, H. W., AND SHIN, S. Y. An example-based approach for facial expression cloning. In *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 167–176, San Diego, California, USA, 26–27 July 2003. Eurographics Association. doi: 10.1145/1185657.1185863. (Cited on pages 1, 2, 7, 9, 14, and 15.)

- RANGARAJAN, A., CHUI, H., MJOISNESS, E., PAPPU, S., DAVACHI, L., GOLDMAN-RAKIC, P., AND DUNCAN, J. A robust point-matching algorithm for autoradiograph alignment. *Medical Image Analysis*, 1(4):379–398, Sept. 1997. doi: 10.1016/S1361-8415(97)85008-6. (Cited on page 21.)
- RUSINKIEWICZ, S. AND LEVOY, M. Efficient variants of the ICP algorithm. In *3DIM '01: Proceedings of the 3rd International Conference on 3D Digital Imaging and Modeling*, pages 145–152, Quebec City, Canada, 28 May–1 June 2001. IEEE Computer Society. doi: 10.1109/IM.2001.924423. (Cited on pages 20, 30, 33, and 72.)
- RUSINKIEWICZ, S., HALL-HOLT, O., AND LEVOY, M. Real-time 3D model acquisition. In *SIGGRAPH' 02: Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques*, pages 438–446. ACM, July 2002. doi: 10.1145/566570.566600. (Cited on page 18.)
- SALVI, J., MATABOSCH, C., FOFI, D., AND FOREST, J. A review of recent range image registration methods with accuracy evaluation. *Image and Vision Computing*, 25(5):578–596, May 2007. doi: 10.1016/j.imavis.2006.05.012. (Cited on page 72.)
- SCOTT, G. L. AND LONGUET-HIGGINS, H. C. An algorithm for associating the features of two images. *Proceedings of the Royal Society B: Biological Sciences*, 244(1309):21–26, Apr. 1991. doi: 10.1098/rspb.1991.0045. (Cited on page 21.)
- SEOL, Y., SEO, J., KIM, P. H., LEWIS, J. P., AND NOH, J. Artist friendly facial animation retargeting. In *SIGGRAPH Asia '11: ACM SIGGRAPH Asia 2011 Papers*, pages 162:1–162:10, Hong Kong, China, 12–15 Dec. 2011. ACM. doi: 10.1145/2024156.2024196. (Cited on pages 1 and 9.)
- SEOL, Y., LEWIS, J. P., SEO, J., CHOI, B., ANJYO, K., AND NOH, J. Spacetime expression cloning for blendshapes. *ACM Transactions on Graphics*, (2):14:1–14:12, Apr. 2012. doi: 10.1145/2159516.2159519. (Cited on pages 2 and 8.)
- SHAMS, R., SADEGHI, P., KENNEDY, R. A., AND HARTLEY, R. I. A survey of medical image registration on multicore and the GPU. *IEEE Signal Processing Magazine*, 27(2):50–60, Mar. 2010. doi: 10.1109/MSP.2009.935387. (Cited on page 18.)
- SHI, J. AND TOMASI, C. Good features to track. In *CVPR '94: Proceedings of the 1994 IEEE Conference on Computer Vision and Pattern Recognition*, pages 593–600, Seattle, WA, USA, 21–23 June 1994. IEEE Computer Society. doi: 10.1109/CVPR.1994.323794. (Cited on page 27.)
- SIFAKIS, E., NEVEROV, I., AND FEDKIW, R. Automatic determination of facial muscle activations from sparse motion capture marker data. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers*, pages 417–425, Los Angeles, California, USA, 31 July–4 Aug. 2005. ACM. doi: 10.1145/1186822.1073208. (Cited on pages 1, 9, 11, and 12.)
- SIFAKIS, E., SELLE, A., ROBINSON-MOSHER, A., AND FEDKIW, R. Simulating speech with a physics-based facial muscle model. In *SCA '06: Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 261–270, Vienna, Austria, 2–4 Sept. 2006. ACM. (Cited on pages 1, 8, 9, and 11.)

- SORKINE, O., COHEN-OR, D., LIPMAN, Y., ALEXA, M., RÖSSL, C., AND SEIDEL, H.-P. Laplacian surface editing. In *SGP '04: Proceedings of the 2nd Eurographics Symposium on Geometry Processing*, pages 175–184, Nice, France, 8–10 July 2004. Eurographics Association. doi: 10.1145/1057432.1057456. (Cited on page 25.)
- SPANIER, E. H. *Algebraic Topology*. Springer, 3rd edition, 1994. ISBN: 0387944265. (Cited on page 83.)
- SUMNER, R. W. *Mesh Modification Using Deformation Gradients*. PhD thesis, MIT, Feb. 2005. (Cited on page 25.)
- SUMNER, R. W. AND POPOVIĆ, J. Deformation transfer for triangle meshes. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, pages 399–405, Los Angeles, California, USA, 8–12 Aug. 2004. ACM. doi: 10.1145/1186562.1015736. (Cited on pages 2, 18, 25, 26, and 54.)
- SUMNER, R. W., ZWICKER, M., GOTSMAN, C., AND POPOVIĆ, J. Mesh-based inverse kinematics. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers*, pages 488–495, Los Angeles, California, USA, 31 July–4 Aug. 2005. ACM. doi: 10.1145/1186822.1073218. (Cited on page 26.)
- SZELISKI, R. *Computer Vision: Algorithms and Applications*. Springer, 2011. ISBN: 978-1-84882-934-3. (Cited on page 18.)
- TAM, G. K., CHENG, Z.-Q., LAI, Y.-K., LANGBEIN, F., LIU, Y., MARSHALL, A. D., MARTIN, R., SUN, X., AND ROSIN, P. Registration of 3d point clouds and meshes: A survey from rigid to non-rigid. *IEEE Transactions on Visualization and Computer Graphics*, 2012. doi: 10.1109/TVCG.2012.310. (in press). (Cited on pages 19 and 22.)
- TERZOPOULOS, D. AND FLEISCHER, K. Deformable models. *The Visual Computer*, 4(6): 306–331, Nov. 1988. doi: 10.1007/BF01908877. (Cited on page 24.)
- TERZOPOULOS, D. AND WATERS, K. Physically-based facial modeling, analysis and animation. *Journal of Visualization and Computer Animation*, 1(2):73–80, 1990. doi: 10.1002/vis.4340010208. (Cited on pages 1, 6, 9, 10, and 11.)
- TERZOPOULOS, D., PLATT, J., BARR, A., AND FLEISCHER, K. Elastically deformable models. In *SIGGRAPH '87: Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, pages 205–214, Anaheim, California, USA, 27–31 July 1987. ACM. doi: 10.1145/37401.37427. (Cited on pages 24, 25, and 36.)
- TERZOPOULOS, D., WITKIN, A., AND KASS, M. Constraints on deformable models: Recovering 3D shape and nonrigid motion. *Artificial Intelligence*, 36(1):91–123, Aug. 1988. doi: 10.1016/0004-3702(88)90080-X. (Cited on page 24.)
- TERZOPOULOS, D., LEE, Y., AND VASILESCU, M. A. O. Model-based and image-based methods for facial image analysis and recognition. In *FGR '04: Proceedings of the 6th IEEE International Conference on Automatic Face and Gesture Recognition*, pages 3–8, Seoul, Korea, 17–19 May 2004. IEEE Computer Society. doi: 10.1109/AFGR.2004.1301501. (Cited on page 8.)

- TRUCCO, E., FUSIELLO, A., AND ROBERTO, V. Robust motion and correspondence of noisy 3-D point sets with missing data. *Pattern Recognition Letters*, 20(9):889–898, Sept. 1999. doi: 10.1016/S0167-8655(99)00055-0. (Cited on page 72.)
- TURK, G. AND LEVOY, M. Zippered polygon meshes from range images. In *SIGGRAPH '94: Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*, pages 311–318, Orlando, Florida, USA, 24–29 July 1994. ACM. doi: 10.1145/192161.192241. (Cited on page 20.)
- UMFPACK. Umfpack: unsymmetric multifrontal sparse lu factorization package. url: <http://www.cise.ufl.edu/research/sparse/umfpack/>. (Cited on pages 38 and 60.)
- VAN KAICK, O., ZHANG, H., HAMARNEH, G., AND COHEN-OR, D. A survey on shape correspondence. *Computer Graphics Forum*, 30(6):1681–1707, Sept. 2011. doi: 10.1111/j.1467-8659.2011.01884.x. (Cited on page 18.)
- VLASIC, D., BRAND, M., PFISTER, H., AND POPOVIĆ, J. Face transfer with multilinear models. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers*, pages 426–433, Los Angeles, California, USA, 31 July–4 Aug. 2005. ACM. doi: 10.1145/1186822.1073209. (Cited on pages 14 and 16.)
- WALDER, C., BREIDT, M., BÜLTHOFF, H., SCHÖLKOPF, B., AND CURIO, C. Markerless 3D face tracking. In *Proceedings of the 31st DAGM Symposium*, volume 5748 of *Lecture Notes in Computer Science*, pages 41–50, Jena, Germany, 9–11 Sept. 2009. doi: 10.1007/978-3-642-03798-6_5. (Cited on pages 14, 27, and 28.)
- WAND, M., JENKE, P., HUANG, Q., BOKELOH, M., GUIBAS, L., AND SCHILLING, A. Reconstruction of deforming geometry from time-varying point clouds. In *SGP '07: Proceedings of the 5th Eurographics Symposium on Geometry Processing*, pages 49–58, Barcelona, Spain, July 2007. Eurographics Association. doi: 10.2312/SGP/SGP07/049-058. (Cited on page 23.)
- WAND, M., ADAMS, B., OVSJAMIKOV, M., BERNER, A., BOKELOH, M., JENKE, P., GUIBAS, L. J., SEIDEL, H.-P., AND SCHILLING, A. Efficient reconstruction of nonrigid shape and motion from real-time 3D scanner data. *ACM Transactions on Graphics*, 28(2): 15:1–15:15, Apr. 2009. doi: 10.1145/1516522.1516526. (Cited on pages 8 and 23.)
- WANG, Y., GUPTA, M., ZHANG, S., WANG, S., GU, X., SAMARAS, D., AND HUANG, P. High resolution tracking of non-rigid motion of densely sampled 3D data using harmonic maps. *International Journal of Computer Vision*, 76(3):283–300, Mar. 2008. doi: 10.1007/s11263-007-0063-y. (Cited on page 27.)
- WATERS, K. A muscle model for animating three-dimensional facial expression. In *SIGGRAPH '87: Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, pages 17–24, Anaheim, California, USA, 27–31 July 1987. ACM. doi: 10.1145/37401.37405. (Cited on pages 1, 6, 9, 10, 11, and 12.)
- WEISE, T., LI, H., VAN GOOL, L., AND PAULY, M. Face/off: live facial puppetry. In *SCA '09: Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 7–16, New Orleans, Louisiana, USA, 1–2 Aug. 2009. ACM. doi: 10.1145/1599470.1599472. (Cited on pages 1, 2, 3, 8, 9, 10, 13, 14, 27, and 28.)

- WEISE, T., BOUAZIZ, S., LI, H., AND PAULY, M. Realtime performance-based facial animation. In *SIGGRAPH '11: ACM SIGGRAPH 2011 Papers*, pages 77:1–77:10, Vancouver, British Columbia, Canada, 7–11 Aug. 2011. ACM. doi: 10.1145/1964921.1964972. (Cited on pages 1, 2, 3, 8, 9, 10, 13, 14, 27, and 28.)
- WEISSTEIN, E. W. Quaternion. url: <http://mathworld.wolfram.com/Quaternion.html>. (Cited on page 63.)
- WILLIAMS, L. Performance-driven facial animation. In *SIGGRAPH '90: Proceedings of the 17th Annual Conference on Computer Graphics and Interactive Techniques*, pages 235–242, Dallas, Texas, USA, 6–10 Aug. 1990. ACM. doi: 10.1145/97879.97906. (Cited on pages 1, 7, 9, 12, 14, and 27.)
- WOLFSON, H. J. AND ROGOUTSOS, I. Geometric hashing: An overview. *IEEE Computational Science and Engineering*, 4(4):10–21, Oct.–Dec. 1997. doi: 10.1109/99.641604. (Cited on page 30.)
- WRIGHT, J., YANG, A. Y., GANESH, A., SATRY, S. S., AND MA, Y. Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):210–227, Feb. 2009. doi: 10.1109/TPAMI.2008.79. (Cited on page 6.)
- XIANG, G., JU, X., HOLT, P. O., AND SHANG, L. Facial expression transferring with a deformable model. In *TPCG '09: EG UK Theory and Practice of Computer Graphics*, pages 117–124, Cardiff, UK, 17–19 June 2009. Eurographics. doi: 10.2312/LocalChapterEvents/TPCG/TPCG09/117-124. (Cited on page 51.)
- XIANG, G., JU, X., AND HOLT, P. O. Automatic 3D facial model and texture reconstruction from range scans. In *AMDO '10: The VI Conference on Articulated Motion and Deformable Objects*, pages 260–269, Andratx, Mallorca, Spain, 7–9 July 2010. Springer. doi: 10.1007/978-3-642-14061-7_25. (Cited on page 29.)
- YILMAZ, A., JAVED, O., AND SHAH, M. Object tracking: A survey. *ACM Computing Surveys*, 38(4):13:1–45, Dec. 2006. doi: 10.1145/1177352.1177355. (Cited on pages 6 and 27.)
- YIN, L., CHEN, X., SUN, Y., WORM, T., AND RALE, M. A high-resolution 3D dynamical facial expression database. In *FGR '08: Proceedings of the 8th IEEE International Conference on Automatic Face and Gesture Recognition*, pages 1–6, Amsterdam, The Netherlands, 17–19 Sept. 2008. IEEE Computer Society. doi: 10.1109/AFGR.2008.4813324. (Cited on pages 3, 28, and 42.)
- YU, L.-F., YEUNG, S.-K., TERZOPOULOS, D., AND CHAN, T. F. Dressup!: Outfit synthesis through automatic optimization. *ACM Transactions on Graphics*, 31(6):134:1–134:14, Nov. 2012. doi: 10.1145/2366145.2366153. (Cited on page 25.)
- YU, Y., ZHOU, K., XU, D., SHI, X., BAO, H., GUO, B., AND SHUM, H.-Y. Mesh editing with poisson-based gradient field manipulation. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, pages 644–651, Los Angeles, California, USA, 8–12 Aug. 2004. ACM. doi: 10.1145/1186562.1015774. (Cited on page 25.)

- ZAYER, R., RÖSSL, C., KAMI, Z., AND SEIDEL, H.-P. Harmonic guidance for surface deformation. In *EUROGRAPHICS '05: Proceedings of the 26th Annual Conference of the Eurographics Association*, pages 601–609, Dublin, Ireland, 29 Aug.–2 Sept. 2005. Eurographics Association. doi: 10.1111/j.1467-8659.2005.00885.x. (Cited on page 25.)
- ZHANG, C. AND ZHANG, Z. A survey of recent advances in face detection. Technical Report MSR-TR-2010-66, June 2010. (Cited on page 6.)
- ZHANG, L., CURLESS, B., AND SEITZ, S. M. Spacetime stereo: Shape recovery for dynamic scenes. In *CVPR '03: Proceedings of the 2003 IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 367–374, Madison, WI, USA, 16–22 June 2003. IEEE Computer Society. doi: 10.1109/CVPR.2003.1211492. (Cited on page 26.)
- ZHANG, L., SNAVELY, N., CURLESS, B., AND SEITZ, S. M. Spacetime faces: High resolution capture for modeling and animation. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, pages 548–558, Los Angeles, California, USA, 8–12 Aug. 2004. ACM. doi: 10.1145/1186562.1015759. (Cited on pages 2, 3, 10, 14, 26, and 27.)
- ZHANG, Q., LIU, Z., GUO, B., TERZOPOULOS, D., AND SHUM, H.-Y. Geometry-driven photorealistic facial expression synthesis. *IEEE Transactions on Visualization and Computer Graphics*, 12(1):48–60, Jan./Feb. 2006. doi: 10.1109/TVCG.2006.9. (Cited on page 13.)
- ZHANG, Z. Iterative point matching for registration of free-form curves. Research Report 1658, INRIA, Mar. 1992. (Cited on pages 19 and 30.)
- ZHAO, W., CHELLAPPA, R., PHILLIPS, P. J., AND ROSENFELD, A. Face recognition: A literature survey. *ACM Computing Surveys*, 35(4):399–458, Dec. 2003. doi: 10.1145/954339.954342. (Cited on page 6.)
- ZINSSER, T., SCHMIDT, J., AND NIEMANN, H. A refined icp algorithm for robust 3-D correspondence estimation. In *ICIP '03: Proceedings of the 2003 IEEE International Conference on Image Processing*, pages 695–698, Barcelona, Catalonia, Spain, 14–18 Sept. 2003. IEEE Computer Society. doi: 10.1109/ICIP.2003.1246775. (Cited on page 72.)
- ZITOVÁ, B. AND FLUSSER, J. Image registration methods: A survey. *Image and Vision Computing*, 21(11):977–1000, Oct. 2003. doi: 10.1016/S0262-8856(03)00137-9. (Cited on page 18.)
- ZOU, G., HUA, J., DONG, M., AND QIN, H. Surface matching with salient keypoints in geodesic scale space. *Computer Animation and Virtual Worlds*, 19(3–4):399–410, Aug. 2008. doi: 10.1002/cav.244. (Cited on page 81.)

This page is intentionally left blank.

Publications

Portions of this thesis were published in the proceedings of the following conferences:

- GUOFU XIANG, XIANGYANG JU, PATRIK O'B. HOLT. Face modelling and tracking from range scans. *BMVC 2010 Postgraduate Workshop*. Aberystwyth, UK. 3 September 2010.
- GUOFU XIANG, XIANGYANG JU, PATRIK O'B. HOLT. Automatic facial expression tracking for 4D range scans. *CGVR '10: The 2010 International Conference on Computer Graphics and Virtual Reality*, pp. 74–78. Las Vegas, Nevada, USA. 12–15 July 2010.
- GUOFU XIANG, XIANGYANG JU, PATRIK O'B. HOLT. Automatic 3D facial model and texture reconstruction from range scans. *AMDO '10: The 6th Conference on Articulated Motion and Deformable Objects*, pp. 260–269. Andratx, Mallorca, Spain. 7–9 July 2010. (*This paper was awarded the third best paper: Best suitable commercial application*)
- GUOFU XIANG, XIANGYANG JU, PATRIK O'B. HOLT, LIN SHANG. Facial expression transferring with a deformable model. *TPCG '09: EG UK Theory and Practice of Computer Graphics*, pp. 117–124. Cardiff, UK. 17–19 June 2009. (*This paper was awarded the best technical paper award in the first round and was also selected as the cover image for the proceedings*)

Some results were also presented at the following symposiums:

- GUOFU XIANG, XIANGYANG JU, PATRIK O'B. HOLT. Facial expression transferring for 3D facial models. *NRP Graduate Student Symposium*, Aberdeen, 20 August 2009

- GUOFU XIANG, XIANGYANG JU, PATRIK O'B. HOLT. An automated approach to establishing dense correspondence for 3D facial models. *BMVA Meeting: Facial Analysis and Animation*, Edinburgh, 10 June 2009.

The author also published the following papers.

- GUOFU XIANG, QINGCHUAN ZHANG, HAOWEN LIU, XIAOPING WU, XIANGYANG JU. Time-resolved deformation measurements of the Portevin-Le Chatelier bands. *Scripta Materialia*, 2007, **56**(8): 721–724.
- GUOFU XIANG, QINGCHUAN ZHANG, HAOWEN LIU, HUIFENG JIANG, XIAOPING WU. Deformation measurements of three types of Portevin-Le Chatelier bands. *Chinese Physics*, 2006, **15**(10): 2378–2384.
- GUOFU XIANG, QINGCHUAN ZHANG, HAOWEN LIU, HUIFENG JIANG, XIAOPING WU. Deformation measurements of serrated yielding shear bands in Al-4%Cu alloy. *Acta Metallurgica Sinica*, 2006, **42**(6): 619–623. (In Chinese)
- LIU HAOWEN, QINGCHUAN ZHANG, JUNYONG LU, GUOFU XIANG, XIAOPING WU. Experimental investigation on the 3D deformation of the nucleation process of PLC shearing band in Al-Cu alloy. *Acta Metallurgica Sinica*, 2006, **42**(9): 925–930. (In Chinese)
- LIU HAOWEN, QINGCHUAN ZHANG, GUOFU XIANG, XIAOPING WU. White-light digital speckle image correlation analysis of the Portevin-Le Chatelier shearing band in multiscale. *Journal of Experimental Mechanics*, 2005, **20**(4): 487–493. (In Chinese)
- GUOFU XIANG, QINGCHUAN ZHANG, HUIFENG JIANG, HAOWEN LIU, SHENGCHENG JIN, XIAOPING WU. Study on the spatial feature of the Portevin-Le Chatelier shearing deformation bands with digital speckle correlation method. *CSEM '05: The 11-th Experimental Mechanics Science Conference of China*. Dalian, China. July 2005. (In Chinese)
- GUOFU XIANG, QINGCHUAN ZHANG, ZHONGJIA CHEN, XIAOPING WU. Microscopic constitutive relation of serrated yielding and its numerical modeling. *The 7-th National Conference on Micro-mechanics and Mechanical Behavior of Material*, Xiamen, China. March 2004. (In Chinese)

Colophon

This thesis was typeset using $\text{\LaTeX} 2_{\epsilon}$ with Hermann Zapf's Palatino typeface family in font size 11pt, text width 369.0pt, and one-and-a-half line spacing. The following \LaTeX packages were used:

amsmath, amssymb, backref, ctable, graphicx, hyperref,
listings, memoir, mathpazo, natbib, prelim2e, xcolor

